Enhancing Text-to-Image Diffusion Transformer via Split-Text Conditioning

Yu Zhang[1] **Jialei Zhou**[1] **Xinchen Li**[1] **Qi Zhang**[1] **Zhongwei Wan**[2]
Duoqian Miao[1*] **Changwei Wang**[1] **Longbing Cao**[3]

[1]Tongji University [2]The Ohio State University [3]Macquarie University

Figure 1: (a) Images generated by MM-DiT 8B-E using different forms of the same caption. Our split-text caption enables the model to notice details, such as ***unripe and partly green***, ***pink ribbon bow***, and display them in the generated image. (b) Comparison of text-to-image generation results between Stable Diffusion v3 and ours. Obviously, ours has better semantic details and mitigates the attribute misbinding such as ***green car***.

Abstract

Current text-to-image diffusion generation typically employs complete-text conditioning. Due to the intricate syntax, diffusion transformers (DiTs) inherently suffer from a comprehension defect of complete-text captions. One-fly complete-text input either overlooks critical semantic details or causes semantic confusion by simultaneously modeling diverse semantic primitive types. To mitigate this defect of DiTs, we propose a novel split-text conditioning framework named DiT-ST. This framework converts a complete-text caption into a split-text caption, a collection of simplified sentences, to explicitly express various semantic primitives and their interconnections. The split-text caption is then injected into different denoising stages of DiT-ST in a hierarchical and incremental manner. Specifically, DiT-ST leverages Large Language Models to parse captions, extracting diverse primitives and hierarchically sorting out and constructing these primitives into a split-text input. Moreover, we partition the diffusion denoising process according to its differential sensitivities to diverse semantic primitive types and determine the appropriate

*Corresponding Author

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

timesteps to incrementally inject tokens of diverse semantic primitive types into input tokens via cross-attention. In this way, DiT-ST enhances the representation learning of specific semantic primitive types across different stages. Extensive experiments validate the effectiveness of our proposed DiT-ST in mitigating the complete-text comprehension defect. Datasets and models are available.

1 Introduction

In recent years, diffusion models [1, 2] have achieved unprecedented breakthroughs [3, 4, 5, 6]. Leveraging the Transformer [7] backbone, the Diffusion Transformer (DiT) [8] has rapidly become the mainstream paradigm for text-to-image generation and achieves impressive performance.

Current DiTs for text-to-image generation mostly employ complete-text captions as conditioning, which are sourced from human-curated datasets or generated by large language models (LLMs) or multimodal large language models (MLLMs). Such complete-text captions, particularly long ones, typically involve complex syntax with intertwined semantic primitives (object, relation, and attribute) and potential redundancy. When dealing with this form of captions, DiTs exhibit an inherent **complete-text comprehension defect**, reflecting in issues such as attribute misbinding [9, 10], style dominance [11, 12], semantic blending [13, 14], and semantic entanglement [15]. The comprehension defect stems from two key aspects: *insufficient semantic analysis* and *premature information exposure*.

On the one hand, *insufficient semantic analysis* arises from multiple factors. For instance, (*i*) text length bottleneck [16]: CLIP text encoder [17] accepts at most 77 tokens, yet effective tokens seldom exceed 20, making it difficult to process detailed text and leading to the truncation or loss of tail information in long captions; (*ii*) softmax competition [18, 19]: tokens compete against each other in the softmax function, causing the representation capability to decrease as token number increases; (*iii*) positional bias [20]: CLIP prioritizes the first item, making it difficult to attend to later items and leading to semantic distortion. These factors collectively hinder models from accurately analyzing semantic primitives and highlighting important semantic information from long complete-text captions. Consequently, models tend to overlook and omit crucial semantic information within a complete-text caption. However, a split-text caption derived by hierarchically sorting out the semantic primitives within the complete-text caption may reduce the syntactical complexity and comprehension difficulty for models, helping improve semantic analysis.

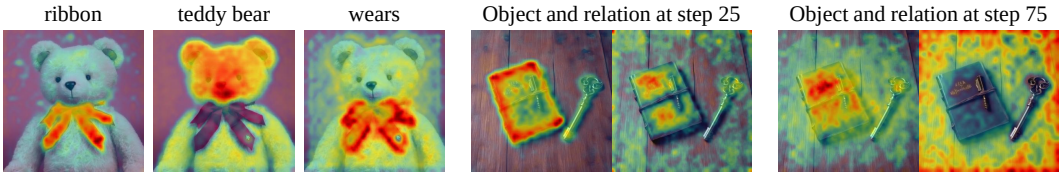


Figure 2: (a) Attention maps [15] for various semantic primitives. Caption: A teddy bear wearing a red ribbon around its neck. Attentions exhibit significant overlap between the object primitive 'ribbon' and relation primitive 'wears', resulting in semantic entanglement. (b) Superimposed attention maps of object primitive type and relation primitive type at denoising timesteps 25 and 75, respectively. Notably, the model focuses more on object primitives during the earlier stage and shifts more attention to relation primitives in the later stage.

On the other hand, the one-time input of a complete-text caption leads to various semantic primitives being modeled simultaneously. Since these primitives cannot be disentangled effectively within the shared representation space, ultimately competing for representation within the same image regions. This is the main reason for phenomena such as semantic entanglement, as illustrated in Figure 2 (a). We summarize this problem as *premature information exposure*, where excessive fine-grained details are prematurely exposed to the model before the model establishes stable primary semantic concepts. Existing studies [21, 22, 23, 24] indicate that diffusion models establish primary semantic concepts during the early denoising stage, while improving fine-grained details in the later stage. Details mainly originate from attribute primitives, while primary semantic concepts derive from object and relation primitives. In addition, as illustrated in Figure 2 (b), different denoising stages have diverse sensitivities of semantic primitive types. We can conclude that the prioritization order for semantic primitive types is object-relation-attribute. Therefore, we propose incrementally injecting diverse types of semantic primitives by prioritization order to improve the proportion of sensitive

primitive types at each stage, enabling the model to enhance the representation learning of a specific sensitive primitive type at the corresponding stage. As discussed above, since a split-text caption is derived by hierarchically sorting out the semantic primitives and possesses a distributed structure, it can effectively group the same type of semantic primitives together and be injected incrementally, naturally helping mitigate premature information exposure.

In this paper, we propose a novel framework, DiT-ST, to enhance the text-to-image DiT via split-text caption. A split-text caption is essentially a hierarchical collection of simplified sentences that explicitly express various semantic primitives and their interconnections, effectively reducing the syntactical complexity to alleviate insufficient semantic analysis and grouping the same type semantic primitives to be injected incrementally to address premature information exposure. Briefly, we first employ LLM [25] for caption parsing to extract various semantic primitives and assemble a hierarchical caption parsing graph. We then construct the hierarchical collection of simplified sentences according to the graph to realize the conversion from complete-text caption to split-text caption. We refine the encoding of the MM-DiT backbone [26], resulting in a DiT-ST with richer semantic level. In addition, we determine appropriate injection timesteps for three semantic primitive types by analyzing cross-attention convergence phenomena [22] and identifying the inflection point in the signal-to-noise ratio curve during denoising. After that, incrementally injecting object-relation-attribute tokens to input tokens via cross-attention to increase the proportions of corresponding sensitive semantic primitive types, further enhancing representation learning of specific semantic primitives at different stages.

Extensive experiments confirm the benefits of our split-text captioning. On GenEval, our DiT-ST-M achieves 0.69 overall accuracy and 11.3% gain over SDv3 Medium. On COCO-5K, it records the highest average CLIPScore (34.09) and the lowest FID (22.11), delivering performance competitive with SDv3.5 Large. All results and visualizations demonstrate that the method is both parameter-efficient and architecture-agnostic, yielding finer detail and stronger semantic fidelity.

2 Related Work

2.1 Diffusion Transformers

Diffusion models [2, 4, 27] reverse a fixed noising process by imitating the denoising trajectory through learned optimization. When integrated with multimodal learning [28, 29, 30, 31, 32] and other techniques [33, 34, 35, 36, 37, 38], they offer a powerful generation framework for visual content [39, 40, 41, 42, 43] and other downstream tasks [44, 45, 46, 47, 48]. Recent advances, Diffusion Transformer [8] replaces U-Net [49] with Transformer [7], enabling its strong capability for long-range dependency modeling, scalability, and flexibility. DiT models, exemplified by DeepFloyd-IF [3], Flux [50], PixArt-α [51] and DreamEngine [52], have successfully achieved remarkable generation performance, making DiT the mainstream paradigm for text-to-image generation.

2.2 Complete-Text Comprehension Defect

Complete-text comprehension defect refers to models' difficulty in effectively analyzing and understanding complete text, stemming from inherent encoder limitations or structural deficiencies, such as text length bottleneck [16], softmax competition [18, 19], positional bias [20], incorrect tokenization [53], and frequency bias [54]. This defect results in suboptimal generation quality and generated content that struggles to faithfully correspond to the prompt. Manifestations of complete-text comprehension defect include attribute misbinding [55], object missing [56], style dominance [11], semantic blending [13], semantic entanglement [15], and etc.

Many existing methods have mitigated specific manifestations. Attend-and-Excite [9] activates dominant token attention to guide the generation process in adherence to the textual prompt. WiCLP [57] and research [10] address attribute misbinding and object omission by refining the text embedding space and reweighting token-level attention to enhance compositional fidelity during generation. DeaDiff [12] mitigates style dominance by employing exclusive subsets of cross-attention layers for disentangling style and semantics. LongAlign [58] enhances semantic coherence by introducing explicit linguistic structures. In contrast, SCoPE [59] operates at the sub-prompt level, beginning with the simplest sub-prompt, then progressively introducing more complex variants and determining injection timesteps using proportional similarity ratios to mitigate alignment degradation.

While these methods indeed mitigate the complete-text comprehension defect from various perspectives, they primarily optimize for specific manifestations. Instead of addressing individual encoder limitations, we target the fundamental cause of the comprehension defect—the inherent inability of models to handle complex syntax within complete-text captions. Convert the form of captions to pursue a comprehensive mitigation strategy for the complete-text comprehension defect.

3 Methodology

The overall framework of DiT-ST is illustrated in Figure 3, which incorporates three key components: Caption Parsing, Hierarchical Caption Input, and Incremental Primitive Injection. The original caption $\mathcal{C}_{CT}$ adopts a complete-text structure. To sort out and refine semantic primitives, DiT-ST employs LLM as for caption parsing, extracting various semantic primitives and assembling them into a caption parsing graph G. According to the relationships among various semantic primitives in graph G, we construct them into a hierarchical collection of simplified sentences—the split-text caption $\mathcal{C}_{ST}$—which is subsequently input to the text-to-image DiT (MM-DiT). During the DiT denoising process, we calculate and determine appropriate timesteps for injecting each of the three semantic primitive types. Following the object-relation-attribute order, we incrementally inject encoded semantic primitive tokens into input tokens via cross-attention thereby enhancing the representation learning of specific semantic primitives across different stages.

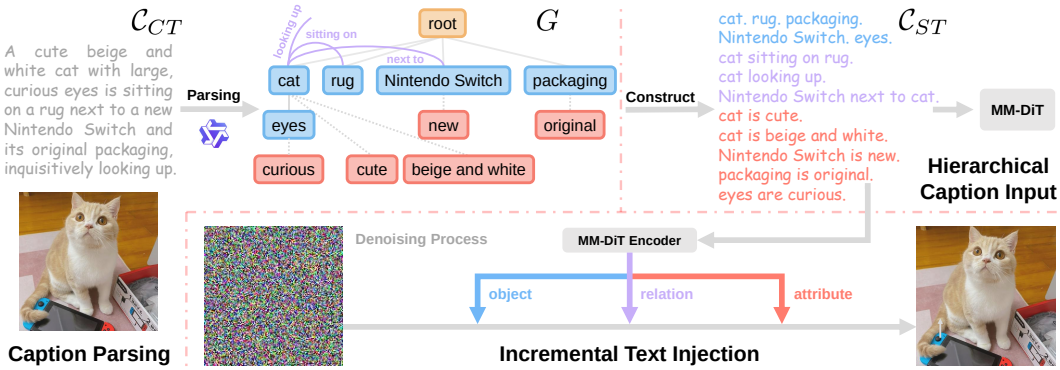


Figure 3: The overall framework of DiT-ST. Three colors represent object, relation and attribute, respectively.

3.1 Caption Parsing

Caption parsing aims to split the complete-text structure of captions, resulting in a caption parsing graph G. The reason why we choose the graph as the target product is that a hierarchical tree-like structure [60] can clearly represent various semantic primitives and their interconnections, naturally resolving existing problems of DiTs in this aspect. Therefore, we leverage a LLM (specifically Qwen-Plus [61] in this work) as a semantic analyzer and graph generator to extract various semantic primitives and assemble these semantic primitives to obtain the caption parsing graph G.

We define a caption parsing graph as $G = (V, E)$, where V denotes the set of N nodes and E denotes the set of edges. Specifically, $V = \{v_i \mid i = 1, 2, \ldots, N\}$, Each node v_i is defined as $v_i = (o_i, \mathcal{A}_i)$, with o_i denoting the i-th object primitive and $\mathcal{A}_i = \{a_j\}$ representing the set of attributes associated with object primitive o_i. $\mathcal{O}$ is the object set, and $\mathcal{A}$ is the attribute set. Relation set $\mathcal{R}$ consists of M semantic relationships: $\mathcal{R} = \{r_k \mid k = 1, 2, \ldots, M\}$. Each edge e is modeled as a labelled triple $e = (v_i, v_j, r_k) \in V \times V \times \mathcal{R}$, the set of edges $E \subseteq V \times V \times \mathcal{R}$.

During the graph assembly, we use the root node to represent the entire caption, and recursively decompose it into various object primitive nodes. Relations between object primitives are represented by edges between nodes. Attributes are represented as child nodes attached to their corresponding object primitive nodes. Therefore, for a given complete-text caption $\mathcal{C}_{CT}$, the process to obtain the caption parsing graph G can be formalized as follows:

$$\mathcal{C}_{CT} \xrightarrow{\text{LLM Parsing}} (\mathcal{O}, \mathcal{R}, \mathcal{A}) \xrightarrow{\text{Graph Assembly}} G. \tag{1}$$

In this way, we convert a complete-text caption into a split and hierarchical caption parsing graph.

3.2 Hierarchical Caption Input

The purpose of hierarchical caption input is to construct a split-text caption based on its caption parsing graph. In addition, in order to make the model better adapt to the split-text caption and enrich the overall semantic level of the input, we also modify the existing text encoding method in text-to-image DiT to better encode the split-text caption. Thus, the hierarchical caption input process consists of two components: split-text caption construction and DiT text encoding refinement.

3.2.1 Split-Text Caption Construction

The hierarchical structure of the caption parsing graph facilitates us to construct a split-text caption, which is a hierarchical collection of simplified sentences. Considering that inherent positional bias in text encoders (e.g., CLIP) is unavoidable during subsequent split-text encoding, we also take into account the order of semantic primitives, i.e., position the important semantic primitives at the beginning to achieve better performance. Specifically, we first rerank object primitives in the object set $\mathcal{O}$ in descending order based on the node degree and the frequency of occurrence in the caption, obtaining a reranked object set $\mathcal{O}'$. We then rerank the primitives in the relation set $\mathcal{R}$ and attribute set $\mathcal{A}$ to ensure that relations or attributes involving object primitives maintain consistency with the object primitive order in $\mathcal{O}'$, resulting in $\mathcal{R}'$ and $\mathcal{A}'$.

Subsequently, based on the reranked sets $\mathcal{O}'$, $\mathcal{R}'$, $\mathcal{A}'$ and following the object-relation-attribute hierarchical order, we generate corresponding simplified sentences for each primitive within these sets. The forms of simplified sentences corresponding to objects, relations, and attributes present respectively `[OBJECT]` object_i, `[RELATION]` object_i relation_k object_j, and `[ATTRIBUTE]` object_i is attribute_i. The collection comprising all these simplified sentences constitutes the split-text caption $\mathcal{C}_{ST}$. The process to construct the split-text caption can be formalized as follows:

$$(\mathcal{O}, \mathcal{R}, \mathcal{A}) \xrightarrow{\text{Acc.to } G \text{ to Rerank}} (\mathcal{O}', \mathcal{R}', \mathcal{A}') \xrightarrow{\text{Construct}} \mathcal{C}_{ST}. \tag{2}$$

3.2.2 DiT Text Encoding Refinement

We employ the current mainstream multimodal diffusion transformer, MM-DiT [26], for text-to-image generation. MM-DiT adopts three text encoders: CLIP-L/14, CLIP-G/14, and T5 XXL. For a split-text caption $\mathcal{C}_{ST}$, after three encoders' text encoding, three token sequences can be obtained: $T^{ST}_{L/14} \in \mathbb{R}^{L \times D_{L/14}}$, $T^{ST}_{G/14} \in \mathbb{R}^{L \times D_{G/14}}$ and $T^{ST}_{T5} \in \mathbb{R}^{L \times D}$, where L is the token sequence length ($L \leq 77$) , $D_{L/14}$, $D_{G/14}$ and D are the dimensions of the three token sequences.

According to the original design, the concatenation of sequences $T^{ST}_{L/14}$ and $T^{ST}_{G/14}$ yields a new token sequence whose still dimension remains smaller than D. Given that the new token sequence must be appended with T^{ST}_{T5} for input into the MM-DiT blocks, the dimension capacity remains underutilized. Therefore, we consider fully utilizing this underutilized dimension capacity by incorporating the complete-text caption $\mathcal{C}_{CT}$ to enrich the overall semantic level of the input. As shown in Figure 4, We supplement it by adding $\mathcal{C}_{CT}$ encoded through T5 XXL and followed by a linear projection, generating the token sequence $T^{CT}_{T5} \in \mathbb{R}^{L \times D'}$, where $D' = D - D_{L/14} - D_{G/14}$. We perform dimensional concatenation and sequence append as follows:

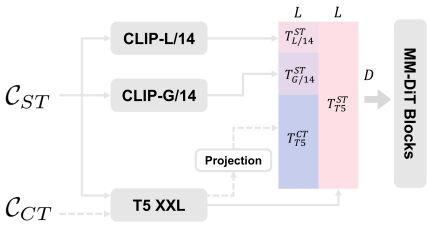


Figure 4: DiT text encoding refinement.

$$T_{concate} = Concate\left(T^{ST}_{L/14}, T^{ST}_{G/14}, T^{CT}_{T5}\right) \in \mathbb{R}^{L \times D}, \tag{3}$$

$$T = Append\left(T_{concate}, T^{ST}_{T5}\right) \in \mathbb{R}^{2L \times D}. \tag{4}$$

Through the above operations, we ultimately construct the input token sequence T which possesses a hierarchical structure of semantic primitives and rich semantic representation.

3.3 Incremental Primitive Injection

As previously discussed, due to different denoising stages have differential sensitivities of semantic primitive types, prematurely exposing fine-grained detail information before the model has established

stable primary semantic concepts is detrimental to generation. We propose incrementally injecting diverse semantic primitive types to enhance differential representation learning for diverse semantic primitive types at different stages. Therefore, incremental primitive injection involves both injection timestep selection and semantic primitive injection.

3.3.1 Injection Timestep Selection

Since diffusion models' prioritization order for diverse semantic primitive types during the denoising process generally follows object-relation-attribute, we should, ideally, identify accurate timesteps for injecting semantic primitives. However, it is practically impossible for several reasons. First, semantic emergence occurs gradually rather than in discrete stages, lacking clear natural boundaries. Second, there are differences across samples, and the semantic emergence process can shift or stretch on the timeline with changes in prompts, resolution, and random seeds. Furthermore, adaptively selecting specific timestep for each sample will incur substantial computational costs. Therefore, we consider statistical rules across samples to identify approximately appropriate injection timesteps.

Determine the injection timestep for attribute primitives. Research [22] indicates that cross-attention outputs converge to a fixed point after several inference steps. This point divides the denoising process into two stages: semantic-planning and fidelity-improving. Since fidelity-improving stage primarily relies on attribute information, we select the timestep corresponding to this convergence point as the injection timestep for attribute primitives. Considering statistical rules across a batch of samples, for a sample, we first define the attention weights output by the h-th attention head in the m-th cross-attention layer at inference step t as $Attn_t^{(m,h)} \in \mathbb{R}^{Q_m \times K_m}$, where Q_m and K_m represent the number of query tokens and key tokens, respectively. Subsequently, we calculate the difference of single-head attention between adjacent inference steps using the Frobenius norm:

$$\Delta_t^{(m,h)} = \frac{||Attn_t^{(m,h)} - Attn_{t-1}^{(m,h)}||_F}{||Attn_{t-1}^{(m,h)}||_F \ + \ \theta}, \tag{5}$$

where θ serves as a numerical stabilizer, typically set to 10^{-8}. Then, we average across all H attention heads and all M cross-attention layers to compute the cross-attention difference for the sample:

$$\Delta_t^{(sample)} = \frac{1}{M}\frac{1}{H}\sum_{m=1}^{M}\sum_{h=1}^{H}\Delta_t^{(m,h)}. \tag{6}$$

By calculating the average cross-attention across all samples in the batch, we obtain the average cross-attention difference $\overline{\Delta}_t$ at t inference step. We employ the moving average method to calculate the average cross-attention $\widehat{\Delta}_t$ across multiple inference steps and set a threshold τ to identify the inference step t^* at which cross-attention converges:

$$\widehat{\Delta}_t = \frac{1}{w}\sum_{k=0}^{w-1}\overline{\Delta}_{t-k}, \qquad t^* = \min\{\, t : \widehat{\Delta}_t < \tau\}, \tag{7}$$

where w denotes the size of the moving window. Thus, the timestep corresponding to inference step t^* is selected as the injection timestep s_{attr} for attribute primitives.

Determine the injection timestep for relation primitives. As previously discussed, given the entire denoising process consists of S timesteps, the first $S - s_{attr}$ timesteps constitute the semantic-planning stage, during which the samples maintain a relatively high signal-to-noise ratio (SNR). Research [21] indicates that semantic concepts are primarily established at a high SNR condition. By analyzing the variation of SNR across timesteps, we observe an initial rapid decline followed by a gradual decrease, as shown in Figure 5. Considering the prioritization order for semantic primitives as object-relation-attribute, intuitively speaking, the inflection point of the SNR within the first $S - s_{attr}$ timesteps can serve as an appropriate injection timestep for relation primitives.

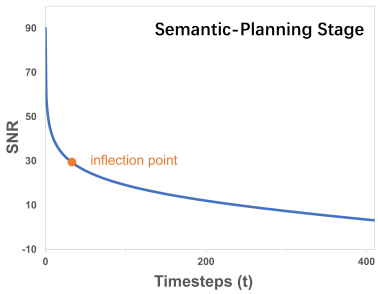


Figure 5: Inflation point of SNR.

We calculate the average SNR $\overline{snr}_t$ at each timestep t across all samples in the batch for the first $S - s_{attr}$ timesteps, and construct a discrete curve function $y_t = g(\overline{snr}_t)$, where $t = 0, 1, \ldots, S - s_{attr} - 1$. We employ the discrete maximum-curvature method to identify the inflection point. First, we calculate the first-order and second-order finite differences as follows:

$$\Delta y_t = \frac{y_{t+1} - y_{t-1}}{\overline{snr}_{t+1} - \overline{snr}_{t-1}}, \qquad \Delta^2 y_t = \frac{y_{t+1} - 2y_t + y_{t-1}}{\left(\frac{1}{2}\left(\overline{snr}_{t+1} - \overline{snr}_{t-1}\right)\right)^2}. \tag{8}$$

Then, we calculate the discrete curvature:

$$\kappa_t = \frac{\left|\Delta^2 y_t\right|}{\left(1 + (\Delta y_t)^2\right)^{3/2}}. \tag{9}$$

In this way, we can obtain the index of inflection point $t^{\#} = \arg\max \kappa_t$, which also is the injection timestep s_{rel} for relation primitives.

Determine the injection timestep for object primitives. Considering the limited number of timesteps before s_{rel}, further partition of these timesteps lacks theoretical and computational evidence. For simplicity, we set the timestep at $t = 0$ as the injection timestep s_{obj} for object primitives.

By identifying the convergence point of cross-attention and the inflection point in SNR curve during denoising, we are able to adaptively determine the appropriate injection timesteps according to statistical rules and characteristics of the sample set.

3.3.2 Semantic Primitive Injection

Semantic primitive enhancement aims to increase the proportion of corresponding semantic primitives to align the semantic primitive type sensitivity of each denoising stage. We employ the cross-attention mechanism and incrementally inject each primitive type $\mathcal{C}^{prim}$ of the split-text caption $\mathcal{C}^{ST}$. $\mathcal{C}^{prim}$ is also encoded by three text encoders (CLIP-L/14, CLIP-G/14, and T5 XXL), obtaining $T^{prim}_{L/14} \in \mathbb{R}^{L \times D_{L/14}}$, $T^{prim}_{G/14} \in \mathbb{R}^{L \times D_{G/14}}$, and $T^{prim}_{T5} \in \mathbb{R}^{L \times D'}$, where $D_{L/14} + D_{G/14} + D' = D$. We then concate these token sequences along the dimension to obtain the primitive injection token sequence T^{prim}, which is used for cross-attention calculation with T to enhance representation learning of the corresponding semantic primitives at the stage.

3.4 Training Strategy

The training objective comprises two components while employing a mixing coefficient λ:

$$\mathcal{L} = \mathcal{L}_{\text{CFM}} + \lambda \mathcal{L}_{\text{attn}}, \tag{10}$$

The first is conditional flow matching loss $\mathcal{L}_{\text{CFM}}$, which supervises denoising prediction, following the setting in MM-DiT. The other is $\mathcal{L}_{\text{attn}}$, which constrains the staged cross-attention by aligning injected semantic primitives with split text while regularizing attention behavior for stable semantic injection. It is composed of three components, weighted with empirically determined ratios of $\alpha = 0.6$, $\beta = 0.25$, and $\eta = 0.15$:

$$\mathcal{L}_{\text{attn}} = \alpha \mathcal{L}_{\text{inject}} + \beta \mathcal{L}_{\text{conv}} + \eta \mathcal{L}_{\text{mutex}}. \tag{11}$$

The first term, $\mathcal{L}_{\text{inject}}$, enforces the alignment between the cross-attention outputs and their target semantics at each activated injection stage:

$$\mathcal{L}_{\text{inject}} = \mathbb{E}_{t,x_0}\left[\sum_{i=1}^{3} \delta_i(t) \left(\text{CrossAttn}_i(c_{\text{base}}, c^{(i)}_{\text{inject}}) - c^{(i)}_{\text{target}}\right)^2\right]. \tag{12}$$

The second term, $\mathcal{L}_{\text{conv}}$, supervises the evolution of the attention signal-to-noise ratio (SNR) to ensure proper convergence timing:

$$\mathcal{L}_{\text{conv}} = \mathbb{E}_t\left[\sum_{i=1}^{3} \left(\text{SNR}^{(i)}_{\text{attn}}(t) - \text{SNR}^{(i)}_{\text{target}}(t)\right)^2\right]. \tag{13}$$

The third term, $\mathcal{L}_{\text{mutex}}$, penalizes spatial overlap between attention maps from different injection stages to maintain independence and prevent semantic interference:

$$\mathcal{L}_{\text{mutex}} = \mathbb{E}_t \left[\sum_{i \neq j} \delta_i(t)\delta_j(t)\text{Overlap}(\text{AttnMap}_i, \text{AttnMap}_j) \right]. \tag{14}$$

Together, these components ensure stable multi-stage semantic injection by enforcing semantic precision, convergence regularity, and spatial exclusivity, which is crucial for achieving fine-grained semantic alignment in Split-Text Conditioning.

4 Experiment

4.1 Experiment Settings

Impementation Details and Datasets To ensure fair comparison beyond the short-text bias of MM-DiT's original training data (CC12M [62] and ImageNet [63]), we adopt 100K samples from the SAM-LLaVA dataset [51], which contains semantically rich, long captions suitable for our split-text method. Extended training on this dataset yields MM-DiT 2B-E and MMDiT 8B-E. Following MM-DiT [26], we use 24 transformer layers with CLIP-L, CLIP-G, and T5-XXL as text encoders. Complete captions are parsed into hierarchical inputs via Qwen-Plus [61], with cross-attention incrementally injected at selected SNR-based denoising steps. Using the same training protocol, we obtain two variants of our models: DiT-ST M (2B) and DiT-ST L (8B). For evaluation, we follow CogView3 [64] and Δ-DiT [65] to construct COCO-5K, assessing performance under varying text lengths and complexities, with results reported from our DiT-ST M and other competing method

Evaluation Metrics We evaluate DiT-ST using both quantitative and qualitative metrics. For quantitative analysis, we adopt GenEval [66] to assess semantic comprehension, along with FID [67] and CLIPScore [68] to measure image quality and text-image alignment.

4.2 Main Results

In this section, we compare our method with MMDiT [26] and its variants as baselines, given architectural similarity and shared training setup. To further evaluate the generalizability of our split-text strategy in mitigating complete-text comprehension detect, we benchmark against competitive methods, including PixArt-α [51], Flux.1 Dev [50], and the state-of-the-art DreamEngine [52].

Effectiveness of Split-text Caption Form Table 1 compares caption utilization forms in terms of CLIPScore and FID, showing that split-text caption leads to consistently better performance. By parsing prompts into organized semantic primitives, our method effectively mitigates the complete-text defect, raises CLIPScore by 7.6 % and lowers FID by 10.2 % relative to the other two caption forms used on MM-DiT 2B-E, while still providing +6.1 % / –9.6 % gains over the LLM-Enhanced caption, underscoreing that the observed gains are primarily attributable to the caption structure, rather than to external LLM intervention.

Table 1: Comparison of the caption forms on COCO-5K.

Caption Form	CLIPScore↑	FID↓
Original Caption	31.68	24.61
LLM-Enhanced Caption	32.13	24.46
Split-Text Caption	**34.09**	**22.11**

Multi-Dimensional Semantic Comprehension Performance Table 2 reports comparisons on the GenEval benchmark, which evaluates six dimensions of semantic understanding: single-object recognition, multi-object reasoning, counting, color accuracy, spatial positioning, and attribute binding. Our method achieves a competitive overall accuracy of 69%, matching the state-of-the-art DreamEngine and closely approaching SDv3.5 Large (71%), despite a $4\times$ smaller parameters. Notably, our model outperforms the SDv3 Medium baseline across all subcategories, with marked advantages in multi-object reasoning(+0.14) and attribute binding(+0.10), indicating stronger semantic comprehension. Compared to Flux.1 Dev and DALL·E 3, which excel in specific areas, our approach offers more balanced and robust performance across all axes due to text-split strategy.

Table 2: Performance on GenEval↑ benchmark,while underlined is the second-best performance.

Method	Parameter	Singel object	Two object	Counting	Colors	Position	Attribute Binding	Overall
PixArt-α	0.6B	0.98	0.50	0.44	0.80	0.08	0.07	0.48
DALL-E 3	-	0.96	0.87	0.47	**0.83**	**0.43**	0.45	0.67
SDv3 Medium	2B	0.98	0.74	0.63	0.67	0.34	0.36	0.62
Flux.1 Dev	12B	0.98	0.81	**0.74**	0.79	0.22	0.45	0.66
SDv3.5 Large	8B	0.98	0.89	0.73	**0.83**	0.34	0.47	**0.71**
Dream Engine	-	**1.00**	**0.94**	0.64	0.81	0.27	**0.49**	0.69
DiT-ST Medium	2B	0.99	0.88	0.70	0.75	0.36	0.46	0.69

Beyond GenEval, we employ CLIPScore and VQAScore [69] on the curated COCO-5K dataset. These two metrics provide a comprehensive evaluation: CLIPScore measures global semantic similarity, while VQAScore leverages a VQA model to directly probe for complex compositional alignment via a probabilistic "Yes/No" query. As results listed in Table 3, DiT-ST Medium attains a CLIPScore of 34.09, eclipsing competing methods by average 8.9 %, and even markedly surpassing SDv3.5 Large (32.74) by about 4.1%. On VQAScore, our DiT-ST Medium achieves 76.12, on par with the larger SDv3.5 Large, while surpassing SDv3 Medium (75.37) and other baselines. These results consistently validate that our split-text conditioning strategy enhances both compositional semantic fidelity and cross-modal alignment, enabling the model to generate images that more accurately capture the intended meaning of complex prompts while maintaining high perceptual quality.

Table 3: Performance of CLIPScore and VQAScore on COCO-5K.

Method	PixArt-α	SDv3 Medium	Flux.1 Dev	SDv3.5 Large	DiT-ST Medium
CLIPScore↑	32.58	31.31	31.52	32.74	**34.09**
VQAScore↑	68.86	75.37	75.19	**76.49**	76.12

Robust Performance to Caption Length Table 4 exposes marked length sensitivity in existing models: PixArt-α benefits from increasing caption length before plateauing, whereas SD-series models peak on mid-length prompts and deteriorate on longer ones (e.g., SD v3 Medium falls to 27.7 in the $[45, 55)$ subband). By contrast, our 2B-parameter model sustains a CLIPScore over 32 across all bins, rises to 35.81 for the longest captions, and secures the highest overall score (34.09), outperforming SDv3 Medium and SD v3.5 Large by 8.9% and 4.1%, respectively. These results highlight the robustness and effectiveness of our split-text conditioning strategy in handling variable-length and semantically complex prompts, enabling more stable text-image alignment across diverse input distributions while remaining parameters effcient.

Table 4: CLIPScore performance comparisons on various caption length in Selected COCO-5K.

Method	Parameter	[10,15]	[15,25)	[25,35)	[35,45)	[45,55]	Average CLIPScore
PixArt-α	0.6B	30.99	31.99	33.26	33.37	33.29	32.58
SDv3 Medium	2B	31.91	32.69	33.37	30.86	27.73	31.31
Flux.1 Dev	12B	30.59	31.16	32.36	31.08	32.83	31.52
SDv3.5 Large	8B	32.17	**32.96**	**34.10**	32.93	31.53	32.74
DiT-ST Medium	2B	**32.28**	32.47	33.84	**34.76**	**35.81**	**34.09**

4.3 Ablation Analysis

Incremental Text Injection Incremental injection improves the text-split pipeline by scheduling semantic primitives at the appropriate timesteps. As Table 5 reports, this schedule rises CLIPScore by 3.9 % and simultaneously reduces FID by 7.3 %, decisively outperforming the single-shot alternative. The progressive delivery deepens semantic grounding, alleviates attention saturation and inter-token interference, enhancing both text alignment and visual fidelity.

Table 5: Comparison of incremental injection under CLIPScore and FID.

Method	CLIPScore↑	FID↓
w/ injection	34.09	22.11
w/o injection	32.81	23.85

Hierarchical Caption Input Hierarchical caption input aligns naturally with the text-split strategy, as it organizes diverse semantic primitive types into a structured format for generation. To assess its effectiveness, we evaluate models under three input settings: original caption, LLM-Enhanced caption, and hierarchical input. As Table 6 shows, merely swapping the caption format to the hierarchical variant lifts CLIPScore by 3.6 % and reduces FID by 3 % for each architecture, with additional 2 % gains over the LLM-Enhanced captions. Crucially, SDv3 Medium enjoys the same absolute improvement as our own model, despite zero fine-tuning, demonstrating that the benefit is inherent to the representation and broadly transferable across models.

Table 6: Comparison among input strategies under CLIP-Score and FID Metrics on COCO-5K.

	Method	CLIPScore↑	FID↓
SDv3 Medium	Original Caption	31.31	24.66
	LLM-Enhanced Caption	31.95	24.46
	Hierarchical Input	**32.42**	**24.05**
DiT-ST Medium	Original Caption	31.68	24.61
	LLM-Enhanced Caption	32.13	24.43
	Hierarchical Input	**32.81**	**23.85**

More Ablation Analysis To further validate the effectiveness of our model and design choices, we conduct additional experiments on the orders of semantic primitive injection, timestep selection, hyperparameter sensitivity, and other ablation studies. Please refer to the appendix for more details.

4.4 Visualization

High-Quality Generation Visualization Figure 6 showcases representative image samples generated by our DiT-ST, with rich visual details and strong semantic alignment across diverse prompts.

(a) (b) (c) (d)

(a) A rustic breakfast withpoached eggs and avocado on sourdough toast, served on a ceramic platewith coffee, bathed in softmorning sunlight.
(b) A vibrant monarch butterfly with orange and black wings perched on a blooming lavender flower, surrounded by other purple blossoms.
(c) At dusk, villagers stroll leisurely along a charming cobblestone street adorned decorated with flower in the rain with their dogs.
(d) A small robin with a bright orange chest perches on arain-drenched branch, surrounded by wet green leaves.

Figure 6: High-quality generation visualization of DiT-ST Large.

More Visualiztion We provide additional visual content, including results of different caption forms (e.g. Figure 1 (a)), performance against SDv3.5 (e.g. Figure 1 (b)) and other competing models as well as our high-quality images at multiple resolutions, which are available in the Appendix.

5 Conclusion

This paper introduces DiT-ST, a split-text conditioning framework that alleviates the complete-text comprehension defect in DiTs. It comprises three components. (*i*) Caption parsing, use LLMs to extract and organize primitives; (*ii*) Hierarchical input construction, construct split-text inputs and enrich input semantics; (*iii*) Incremental primitive injection, inject different primitives into appropriate denoising stages to improve stage-specific representation learning. Extensive experiments demonstrate the effectiveness of our split-text caption design and the excellent performance of our proposed DiT-ST. We hope this work will offer some inspiration to the diffusion community.

Acknowledgement

This work is supported by the National Natural Science Foundation of China (No. 62576251, No. 62376198, and No. 62576247), the National Key Research and Development Program of China (No. 2022YFB3104700), and the Fundamental Research Funds for the Central Universities. The authors would like to thank Tianyu Wang and Jiu for generous support.

References

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

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

[3] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in neural information processing systems*, 35:36479–36494, 2022.

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

[5] Ling Yang, Zhilong Zhang, Yang Song, Shenda Hong, Runsheng Xu, Yue Zhao, Wentao Zhang, Bin Cui, and Ming-Hsuan Yang. Diffusion models: A comprehensive survey of methods and applications. *ACM Computing Surveys*, 56(4):1–39, 2023.

[6] Hui Shen, Jingxuan Zhang, Boning Xiong, Rui Hu, Shoufa Chen, Zhongwei Wan, Xin Wang, Yu Zhang, Zixuan Gong, Guangyin Bao, et al. Efficient diffusion models: A survey. *arXiv preprint arXiv:2502.06805*, 2025.

[7] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[8] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4195–4205, 2023.

[9] Hila Chefer, Yuval Alaluf, Yael Vinker, Lior Wolf, and Daniel Cohen-Or. Attend-and-excite: Attention-based semantic guidance for text-to-image diffusion models. *ACM transactions on Graphics (TOG)*, 42(4):1–10, 2023.

[10] Weixi Feng, Xuehai He, Tsu-Jui Fu, Varun Jampani, Arjun Akula, Pradyumna Narayana, Sugato Basu, Xin Eric Wang, and William Yang Wang. Training-free structured diffusion guidance for compositional text-to-image synthesis. *arXiv preprint arXiv:2212.05032*, 2022.

[11] Wen Li, Muyuan Fang, Cheng Zou, Biao Gong, Ruobing Zheng, Meng Wang, Jingdong Chen, and Ming Yang. Styletokenizer: Defining image style by a single instance for controlling diffusion models. In *European Conference on Computer Vision*, pages 110–126. Springer, 2024.

[12] Tianhao Qi, Shancheng Fang, Yanze Wu, Hongtao Xie, Jiawei Liu, Lang Chen, Qian He, and Yongdong Zhang. Deadiff: An efficient stylization diffusion model with disentangled representations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8693–8702, 2024.

[13] Omri Avrahami, Ohad Fried, and Dani Lischinski. Blended latent diffusion. *ACM transactions on graphics (TOG)*, 42(4):1–11, 2023.

[14] Lorenzo Olearo, Giorgio Longari, Simone Melzi, Alessandro Raganato, and Rafael Peñaloza. How to blend concepts in diffusion models. *arXiv preprint arXiv:2407.14280*, 2024.

[15] Raphael Tang, Linqing Liu, Akshat Pandey, Zhiying Jiang, Gefei Yang, Karun Kumar, Pontus Stenetorp, Jimmy Lin, and Ferhan Ture. What the daam: Interpreting stable diffusion using cross attention. *arXiv preprint arXiv:2210.04885*, 2022.

[16] Beichen Zhang, Pan Zhang, Xiaoyi Dong, Yuhang Zang, and Jiaqi Wang. Long-clip: Unlocking the long-text capability of clip. In *European Conference on Computer Vision*, pages 310–325. Springer, 2024.

[17] 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.

[18] Zhilin Yang, Zihang Dai, Ruslan Salakhutdinov, and William W Cohen. Breaking the softmax bottleneck: A high-rank rnn language model. *arXiv preprint arXiv:1711.03953*, 2017.

[19] Amita Kamath, Jack Hessel, and Kai-Wei Chang. Text encoders bottleneck compositionality in contrastive vision-language models. *arXiv preprint arXiv:2305.14897*, 2023.

[20] Reza Abbasi, Ali Nazari, Aminreza Sefid, Mohammadali Banayeeanzade, Mohammad Hossein Rohban, and Mahdieh Soleymani Baghshah. Clip under the microscope: A fine-grained analysis of multi-object representation. *arXiv preprint arXiv:2502.19842*, 2025.

[21] Jooyoung Choi, Jungbeom Lee, Chaehun Shin, Sungwon Kim, Hyunwoo Kim, and Sungroh Yoon. Perception prioritized training of diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11472–11481, 2022.

[22] Wentian Zhang, Haozhe Liu, Jinheng Xie, Francesco Faccio, Mike Zheng Shou, and Jürgen Schmidhuber. Cross-attention makes inference cumbersome in text-to-image diffusion models. *arXiv e-prints*, pages arXiv–2404, 2024.

[23] Grace Luo, Lisa Dunlap, Dong Huk Park, Aleksander Holynski, and Trevor Darrell. Diffusion hyperfeatures: Searching through time and space for semantic correspondence. *Advances in Neural Information Processing Systems*, 36:47500–47510, 2023.

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

[25] Baichuan Inc. Qwen: A scalable and multilingual language model family, 2024. `https://huggingface.co/Qwen/Qwen-14B-Plus`.

[26] Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, et al. Scaling rectified flow transformers for high-resolution image synthesis. In *Forty-first international conference on machine learning*, 2024.

[27] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *CoRR*, abs/2010.02502, 2020.

[28] Yu Zhang, Qi Zhang, Zixuan Gong, Yiwei Shi, Yepeng Liu, Duoqian Miao, Yang Liu, Ke Liu, Kun Yi, Wei Fan, et al. Mlip: Efficient multi-perspective language-image pretraining with exhaustive data utilization. *arXiv preprint arXiv:2406.01460*, 2024.

[29] Bokun Wang, Yang Yang, Xing Xu, Alan Hanjalic, and Heng Tao Shen. Adversarial cross-modal retrieval. In *Proceedings of the 25th ACM international conference on Multimedia*, pages 154–162, 2017.

[30] Yepeng Liu, Yiren Song, Hai Ci, Yu Zhang, Haofan Wang, Mike Zheng Shou, and Yuheng Bu. Image watermarks are removable using controllable regeneration from clean noise. *arXiv preprint arXiv:2410.05470*, 2024.

[31] Rongtao Xu, Jiguang Zhang, Jiaxi Sun, Changwei Wang, Yifan Wu, Shibiao Xu, Weiliang Meng, and Xiaopeng Zhang. Mrftrans: Multimodal representation fusion transformer for monocular 3d semantic scene completion. *Information Fusion*, 111:102493, 2024.

[32] Zixuan Gong, Qi Zhang, Guangyin Bao, Lei Zhu, Yu Zhang, Ke Liu, Liang Hu, and Duoqian Miao. Lite-mind: Towards efficient and robust brain representation learning. In *ACM Multimedia 2024*, 2024.

[33] Zhongwei Wan, Xinjian Wu, Yu Zhang, Yi Xin, Chaofan Tao, Zhihong Zhu, Xin Wang, Siqi Luo, Jing Xiong, and Mi Zhang. D2o: Dynamic discriminative operations for efficient generative inference of large language models. *arXiv preprint arXiv:2406.13035*, 2024.

[34] Kexue Fu, Mingzhi Yuan, Changwei Wang, Weiguang Pang, Jing Chi, Manning Wang, and Longxiang Gao. Dual focus-attention transformer for robust point cloud registration. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 11769–11778, 2025.

[35] Zhongwei Wan, Zhihao Dou, Che Liu, Yu Zhang, Dongfei Cui, Qinjian Zhao, Hui Shen, Jing Xiong, Yi Xin, Yifan Jiang, et al. Srpo: Enhancing multimodal llm reasoning via reflection-aware reinforcement learning. *arXiv preprint arXiv:2506.01713*, 2025.

[36] Jinzhou Lin, Han Gao, Xuxiang Feng, Rongtao Xu, Changwei Wang, Man Zhang, Li Guo, and Shibiao Xu. Advances in embodied navigation using large language models: A survey. *arXiv preprint arXiv:2311.00530*, 2023.

[37] Yu Zhang, Yepeng Liu, Duoqian Miao, Qi Zhang, Yiwei Shi, and Liang Hu. Mg-vit: a multi-granularity method for compact and efficient vision transformers. *Advances in Neural Information Processing Systems*, 36:69328–69347, 2023.

[38] Yepeng Liu and Yuheng Bu. Adaptive text watermark for large language models. *arXiv preprint arXiv:2401.13927*, 2024.

[39] Prafulla Dhariwal and Alex Nichol. Diffusion models beat gans on image synthesis. *CoRR*, abs/2105.05233, 2021.

[40] Jonathan Ho, Tim Salimans, Alexey Gritsenko, William Chan, Mohammad Norouzi, and David J. Fleet. Video diffusion models, 2022.

[41] Weimin Qiu, Jieke Wang, and Meng Tang. Self-cross diffusion guidance for text-to-image synthesis of similar subjects. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 23528–23538, 2025.

[42] Jinjin Zhang, Qiuyu Huang, Junjie Liu, Xiefan Guo, and Di Huang. Diffusion-4k: Ultra-high-resolution image synthesis with latent diffusion models. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 23464–23473, 2025.

[43] Junyang Chen, Jinshan Pan, and Jiangxin Dong. Faithdiff: Unleashing diffusion priors for faithful image super-resolution. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 28188–28197, 2025.

[44] Zixuan Gong, Guangyin Bao, Qi Zhang, Zhongwei Wan, Duoqian Miao, Shoujin Wang, Lei Zhu, Changwei Wang, Rongtao Xu, Liang Hu, Ke Liu, and Yu Zhang. Neuroclips: Towards high-fidelity and smooth fmri-to-video reconstruction. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang, editors, *Advances in Neural Information Processing Systems*, volume 37, pages 51655–51683. Curran Associates, Inc., 2024.

[45] Bencheng Liao, Shaoyu Chen, Haoran Yin, Bo Jiang, Cheng Wang, Sixu Yan, Xinbang Zhang, Xiangyu Li, Ying Zhang, Qian Zhang, and Xinggang Wang. Diffusiondrive: Truncated diffusion model for end-to-end autonomous driving. *arXiv preprint arXiv:2411.15139*, 2024.

[46] Nanxin Chen, Yu Zhang, Heiga Zen, Ron J Weiss, Mohammad Norouzi, and William Chan. Wavegrad: Estimating gradients for waveform generation. *arXiv preprint arXiv:2009.00713*, 2020.

[47] Kunpeng Qiu, Zhiqiang Gao, Zhiying Zhou, Mingjie Sun, and Yongxin Guo. Noise-consistent siamese-diffusion for medical image synthesis and segmentation, 2025.

[48] Xiaomin Li, Mykhailo Sakevych, Gentry Atkinson, and Vangelis Metsis. Biodiffusion: A versatile diffusion model for biomedical signal synthesis, 2024.

[49] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation, 2015.

[50] Black Forest Labs. Flux. `https://github.com/black-forest-labs/flux`, 2024.

[51] Junsong Chen, Jincheng Yu, Chongjian Ge, Lewei Yao, Enze Xie, Yue Wu, Zhongdao Wang, James Kwok, Ping Luo, Huchuan Lu, and Zhenguo Li. Pixart-α: Fast training of diffusion transformer for photorealistic text-to-image synthesis, 2023.

[52] Liang Chen, Shuai Bai, Wenhao Chai, Weichu Xie, Haozhe Zhao, Leon Vinci, Junyang Lin, and Baobao Chang. Multimodal representation alignment for image generation: Text-image interleaved control is easier than you think, 2025.

[53] Dixuan Wang, Yanda Li, Junyuan Jiang, Zepeng Ding, Guochao Jiang, Jiaqing Liang, and Deqing Yang. Tokenization matters! degrading large language models through challenging their tokenization. *arXiv preprint arXiv:2405.17067*, 2024.

[54] Richard Diehl Martinez, Zébulon Goriely, Andrew Caines, Paula Buttery, and Lisa Beinborn. Mitigating frequency bias and anisotropy in language model pre-training with syntactic smoothing. *arXiv preprint arXiv:2410.11462*, 2024.

[55] Tsung-Han Wu, Long Lian, Joseph E Gonzalez, Boyi Li, and Trevor Darrell. Self-correcting llm-controlled diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6327–6336, 2024.

[56] Tuna Han Salih Meral, Enis Simsar, Federico Tombari, and Pinar Yanardag. Conform: Contrast is all you need for high-fidelity text-to-image diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9005–9014, 2024.

[57] Arman Zarei, Keivan Rezaei, Samyadeep Basu, Mehrdad Saberi, Mazda Moayeri, Priyatham Kattakinda, and Soheil Feizi. Improving compositional attribute binding in text-to-image generative models via enhanced text embeddings, 2025.

[58] Luping Liu, Chao Du, Tianyu Pang, Zehan Wang, Chongxuan Li, and Dong Xu. Improving long-text alignment for text-to-image diffusion models, 2025.

[59] Ketan Suhaas Saichandran, Xavier Thomas, Prakhar Kaushik, and Deepti Ghadiyaram. Progressive prompt detailing for improved alignment in text-to-image generative models, 2025.

[60] Ziwei Yao, Ruiping Wang, and Xilin Chen. Hifi-score: Fine-grained image description evaluation with hierarchical parsing graphs. In *European Conference on Computer Vision*, pages 441–458. Springer, 2024.

[61] An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, et al. Qwen2. 5 technical report. *arXiv preprint arXiv:2412.15115*, 2024.

[62] Soravit Changpinyo, Piyush Sharma, Nan Ding, and Radu Soricut. Conceptual 12m: Pushing web-scale image-text pre-training to recognize long-tail visual concepts, 2021.

[63] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009.

[64] Wendi Zheng, Jiayan Teng, Zhuoyi Yang, Weihan Wang, Jidong Chen, Xiaotao Gu, Yuxiao Dong, Ming Ding, and Jie Tang. Cogview3: Finer and faster text-to-image generation via relay diffusion, 2024.

[65] Pengtao Chen, Mingzhu Shen, Peng Ye, Jianjian Cao, Chongjun Tu, Christos-Savvas Bouganis, Yiren Zhao, and Tao Chen. δ-dit: A training-free acceleration method tailored for diffusion transformers, 2024.

[66] Dhruba Ghosh, Hanna Hajishirzi, and Ludwig Schmidt. Geneval: An object-focused framework for evaluating text-to-image alignment, 2023.

[67] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium, 2018.

[68] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision, 2021.

[69] Zhiqiu Lin, Deepak Pathak, Baiqi Li, Jiayao Li, Xide Xia, Graham Neubig, Pengchuan Zhang, and Deva Ramanan. Evaluating text-to-visual generation with image-to-text generation. In *European Conference on Computer Vision*, pages 366–384. Springer, 2024.

A Dataset Details

We further introduce the datasets used in our work from two aspects: training and evaluation.

For training, we observe that existing datasets used by baseline models MM-DiT [26] are suboptimal for complex caption-conditioned generation. Specifically, datasets like CC12M [62] have relatively short captions (average length 10.3), and ImageNet [63] provides only class-level supervision. Inspired by Pixart-α [51], which employs LLaVA to refine raw captions into high-information-density text, we adopt the SAM-LLaVA dataset provided by Pixart-α to enhance textual supervision. We select 100K pairs from SAM-LLaVA (available in our released CSV files) to further train MM-DiT 2B and 8B, aiming to improve their capacity on long-text generation and ensure fair comparisons.

For evaluation, we construct COCO-5K, following settings from CogView3 [64] and Δ-DiT [65]. It comprises 5,000 image-text pairs sampled across multiple caption-length intervals to assess performance under varying textual complexity. Additionally, we report gFID scores based on COCO-30K to assess distribution-level image fidelity. Complete results are provided in Appendix D.

B Implementation Details

In terms of implementation, we focus on the settings of key thresholds and corresponding code-level realizations. We set the moving average window size w to 3, allowing each $\widehat{\Delta}_t$ to consider the current and two preceding steps for better capturing the SNR trend. To ensure numerical stability when computing normalized differences, we set $\theta = 10^{-8}$. For identifying the convergence point, we use a convergence threshold $\tau = 10^{-4}$. During inference, we select 40 denoising steps from the full diffusion range (0–1000) via uniform random sampling. When determining the injection layers, we allow flexible injection intervals. For example, around timestep 50, we inject attribute primitives within a ± 10 timestep range, while around timestep 400, we permit a broader window of ± 40 timesteps, reflecting the varying stability of attention dynamics at different diffusion stages.

C Further Evaluation and Model Comparison

To reduce metric variance and ensure fair model comparison, we construct a fixed 30K image-text subset from COCO for consistent evaluation across architectures and training scales.We evaluate and compare our method against several representative baselines, including SDv3 Medium, SDv3.5 Large, Flux .1 Dev, Pixart-α. The evaluation results, in terms of FID, are summarized in Table 7.On the fixed COCO-30K subset, our DiT-ST Large model **obtain the lowest FID scores**. At the 2B scale, DiT-ST Medium reaches 18.78, narrowly beating SD v3 Medium (18.82) by 0.2 %. With 8B parameters, DiT-ST Large achieves 17.16, improving on SD v3.5 Large (17.31) by 0.9 %, a modest yet consistent gain that shows split-text conditioning still reduces late-stage artefacts even on short COCO captions. Flux .1 Dev (32.10) and PixArt-α (27.35) post FIDs about 40% higher than DiT-ST Large. Flux, tuned for fast continuous-token sampling and lacking COCO fine-tuning, yields texture artefacts; PixArt-α, trained on long, stylised captions, mismatches short, photorealistic domain. Thus, DiT-ST equals or surpasses SD baselines and clearly outperforms models trained on divergent data.

Table 7: Performance of FID on COCO-30K.

Method	PixArt-α	SDv3 Medium	Flux.1 Dev	DiT-ST Medium	SDv3.5 Large	DiT-ST Large
Param.	0.6B	2B	12B	2B	8B	8B
FID↓	27.35	18.82	32.10	18.78	17.31	**17.16**

D Ablation on Injection Order of Semantic Primitives.

In this experiment, we investigate how varying the injection order of semantic primitives, namely object, relation, and attribute, affects model performance. While our default configuration follows an object–relation–attribute order based on semantic granularity and generation stability, we test alternative permutations to examine the sensitivity of DiT-ST Medium to ordering strategies.Table 8 reveals the cost of premature information exposure. Advancing attribute tokens from their intended late-stage slot to attribute-relation-object or attribute-object-relation cuts CLIPScore to 29.41 and

31.10, drops of 14 % and 9 % from the default order. Early injection forces fine-grained colour and texture cues into an unstable attention landscape; later denoising steps overwrite or mis-bind these details, sharply reducing semantic fidelity.

Table 8: Comparison of different injection order for semantic primitives.

Order	CLIPScore↑
attribute-relation-object	29.41
attribute-object-relation	31.10
relation-object-attribute	32.52
relation-attribute-object	30.16
object-attribute-relation	30.79
object-relation-attribute (our setting)	**34.09**

E Ablation on step selection strategy for semantic primitive injection.

We further investigate the impact of injection timestep choices for different semantic primitive types in the diffusion denoising process. Given that semantic primitives differ in abstraction level and generation dependency, we design a progressive injection schedule tailored to each type:

- Object tokens are injected at the early stages of denoising (timesteps 0, 10, and 20), where the model primarily focuses on establishing global layout and coarse structural elements. We experiment with these positions to compare their impact and identify the most effective timing for guiding the foundational composition of the generated image.
- Relation tokens are injected during mid-stage denoising. Specifically, we begin with timestep 25 as the midpoint of [0, 50], and extend to timestep 75 with five evenly spaced steps (30, 40, 50, 60, 70). We experiment with these positions to assess their impact and determine the optimal timing for enhancing spatial and compositional coherence.
- Attribute tokens are injected during the transition from semantic interpretation to visual detail refinement, when the model shifts focus from global structure to fine-grained features. To study their impact on visual modeling, we take the midpoint of the [50, 400] range as reference and select timesteps at 100-step intervals—specifically 200, 300, 400, 500, and 600—for evaluation.

This step allocation strategy reflects the assumption that lower-level semantics (e.g., object identity) should be introduced early to guide coarse synthesis, while higher-level or localized attributes benefit from later-stage injection when image fidelity and semantic detail are resolved.

As shown in Table 9, deviating from the default object-relation-attribute (O–R–A) schedule markedly impairs text–image alignment. Postponing object injection to $t = 10$–20 lowers CLIPScore by roughly 5–6 %, underscoring the need for early global cues. Advancing relation tokens to $t = 30$ reduces the score by about 4 %, as premature relational reasoning diverts attention from still-forming objects. The most severe loss, nearly 8 %, occurs when attribute tokens are introduced at $t = 200$; exposing fine-grained details before the scene is stabilised disrupts subsequent refinement as claimed in our main paper the question of premature information exposure. In aggregate, mis-timed injections can **degrade alignment by average 5%**, confirming that stage-aware scheduling is critical for maintaining semantic fidelity.

Table 9: Comparison of Different Step Selection Strategy by CLIPScore(CS).

Step	O(10)	O(20)	R(30)	R(40)	R(60)	R(70)	A(200)	A(300)	A(500)	A(600)	Default
CS↑	32.71	32.24	32.57	32.73	32.76	32.45	31.26	31.75	<u>32.96</u>	32.83	**34.09**

F Ablation on Sliding Window Size

Table 10 shows that CLIPScore peaks at 34.09Our when the SNR-smoothing window is set to $w = 3$.A narrow window ($w = 1$) amplifies local noise, shifting the detected inflection point and trimming the score to 33.16 (-2.7 %). Expanding to $w = 2$ alleviates some volatility, yet still lags

the optimum at 33.55 (-1.6 %). Oversmoothing with $w = 4$ and $w = 5$ postpones relation-token injection; scores decline to 33.72 (-1.1 %) and 33.43 (-1.9 %), respectively. These results confirm that an intermediate window is essential—wide enough to suppress spurious curvature spikes yet narrow enough to capture the true SNR transition—thereby yielding the most consistent relation-level guidance and highest text–image alignment.

Table 10: Comparison of different sizes of sliding window.

Window Size	CLIPScore↑
1	33.16
2	33.55
3 (our setting)	**34.09**
4	33.72
5	33.43

G Ablation on Text Encoding Refinement.

As detailed in Section 3.2.2, our design includes a refinement for the classical text encoder of MM-DiT. Specifically, we fully utilize the previously wasted dimension capacity by incorporating the complete caption to enrich the overall semantic information of the input. To further validate the effectiveness of this refinement, we conduct an ablation experiment on our DiT-ST Medium model, trained for 5 epochs on 50K SAM-LLaVA dataset. The results are presented in Table 11. Without text encoding refinement, the model performance has declines of 0.62% in CLIPScore and 0.54% in FID metrics. The result indicate that although the improvement from adding complete text captions is modest, it indeed contributes positively to model performance without introducing additional computational or architectural overhead.

Table 11: Comparison of w/ and w/o Text Encoding Refinement.

Method	CLIPScore↑	FID↓
w/ Text Encoding Refinement	**32.33**	**24.18**
w/o Text Encoding Refinement	32.13	24.31

H Limitation

The injection timestep selection operates at the batch level rather than the sample level. This is because sample-level adaptation would significantly increase computational cost and reduce inference efficiency, coupled with the current lack of research on sample-level adaptation in the conditioning of denoising processes. Therefore, this study adopts a batch-level adaptation design as a compromise.

The split-text caption may weaken cross-primitive semantic dependency. Some special long-range dependencies (e.g., irony) could be split, leading to biased semantic understanding in the model. To mitigate this issue, DiT-ST introduces the text encoding refinement design, which incorporates original caption to compensate for the loss. Furthermore, existing works in the same field have not explored evaluating the semantic fidelity of special cases such as irony.

The evaluation metrics lack human subjective assessment. The evaluation of DiT-ST primarily relies on CLIPScore and FID, aligning with other works in the same filed. These metrics lack human perception, particularly in fine-grained details (e.g., color tone, detail consistency). To compensate for this limitation, similar to other prior works, we provide extensive visualizations to facilitate subjective evaluation by readers.

I More Visual Comparison Across Caption Forms

Original Caption (Complete-Text) | LLM-Enhanced Caption (Complete-Text) | Our Caption (Split-Text)

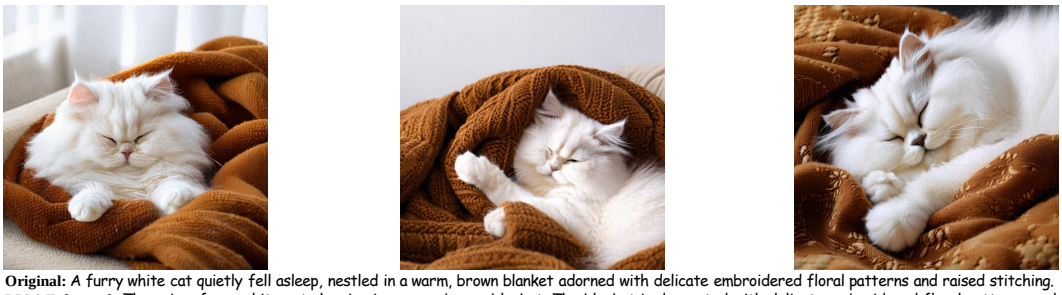
Original: A quiet beach with soft sand, scattered white seashells. One small bird stands alone near the water, its feet in the shallow waves, looking down at its reflection. Gentle waves roll in behind it.
LLM-Enhanced: There is a quiet beach with soft sand and scattered white seashells. A small bird stands alone near the water, looking at its reflection.
Our: [OBJECT] beach. sand. seashells. bird. waves. [RELATION] beach has sand and seashells. bird stands near water. feet are in waves. bird looks at its reflection. waves roll in behind the bird. [ATTRIBUTE] The beach is quiet. The sand is soft. The seashells are white and scattered. The bird is small and alone. The waves are gentle and shallow.

Original: In a quiet forest filled with tall pine trees, a young deer with dots on its back stands still in the soft moss underfoot.
LLM-Enhanced: There are a quiet forest, tall pine trees and a young deer. The deer stands still on soft moss floor.
Our: [OBJECT] forest. trees. deer. moss. [RELATION] forest has tall pine trees. deer stands in moss. [ATTRIBUTE] The forest is quiet. The trees are tall and pine. The deer is young and with dots. The moss is soft.

Original: A yellow school bus drove through a rainy city street at night. The damp road reflects the yellow shop lights on roadside, heavy rain is falling.
LLM-Enhanced: There is a yellow school bus driving through a rainy city street at night. The wet road reflects the yellow lights of nearby shop lights. The rain falls heavily and steadily.
Our: [OBJECT] school bus. street. shop. lights. rain. [RELATION] bus drives on damp street. shop has lights. street reflects lights. [ATTRIBUTE] The bus is yellow. The street is damp. The lights are yellow. The rain is heavy.

Original: At a morning café, a man was flipping through newspaper by the window, with warm sunlight shining across the wooden table and a cup of coffee beside him.
LLM-Enhanced: There is a man sitting by the window in a morning café, reading newspaper. Warm sunlight shines across the wooden table, and a cup of coffee beside him.
Our: [OBJECT] café. man. newspaper. window. sunlight. table. coffee. [RELATION] man sits by the window. man flips through newspaper. sunlight shines on the table. coffee is beside the man. [ATTRIBUTE] The café is morning. The table is wooden. The sunlight is warm.

Original Caption (Complete-Text) | LLM-Enhanced Caption (Complete-Text) | Our Caption (Split-Text)



Original: A furry white cat quietly fell asleep, nestled in a warm, brown blanket adorned with delicate embroidered floral patterns and raised stitching.
LLM-Enhanced: There is a furry white cat sleeping in a warm brown blanket. The blanket is decorated with delicate embroidered floral patterns and raised stitching.
Our: [OBJECT] cat. blanket. patterns. stitching. [RELATION] cat sleeps in blanket. blanket is adorned with floral patterns. blanket has raised stitching. [ATTRIBUTE] The cat is furry and white. The blanket is warm and brown. The floral patterns are delicate and embroidered. The stitching is raised.

Original: A silver pen without its cap rests on a wooden table beside a blank white spiral-bound notebook. A few other notebooks also lie nearby.
LLM-Enhanced: There is a silver pen without cap resting on a wooden table next to a blank white spiral-bound notebook. Several other notebooks are scattered nearby.
Our: [OBJECT] pen. table. notebook. [RELATION] pen rests on table. pen is beside notebook. other notebooks lie nearby. [ATTRIBUTE] The pen is silver and uncapped. The table is wooden. The notebook is blank, white, and spiral-bound.

Original: A crispy fried chicken burger with lettuce, pickles, and cream sauce on top, placed beside a container of golden French fries on a wooden table.
LLM-Enhanced: There is a crispy fried chicken burger topped with lettuce, pickles, and cream sauce, sitting next to a container of golden French fries on a wooden table.
Our: [OBJECT] burger. lettuce. pickles. cream sauce. fries. container. table. [RELATION] burger has lettuce, pickles, and cream sauce. burger is beside fries. fries are in a container. burger and fries are on the table. [ATTRIBUTE] The burger is crispy and fried. The lettuce, pickles and cream sauce is on top. The fries are golden. The table is wooden.

Original: A healthy meal with two egg yolk facing fried eggs, fresh half cut avocados, and a few slices of toasted whole wheat bread, lightly seasoned with herb.
LLM-Enhanced: There is a healthy meal consisting of two sunny-side-up eggs, fresh halved avocados, and a few slices of toasted whole wheat bread, lightly seasoned with herb.
Our: [OBJECT] meal. eggs. avocados. bread. herb. [RELATION] meal includes eggs, avocados, and bread. meal is seasoned with herb. [ATTRIBUTE] The meal is healthy. The eggs are fried and yolks up. The avocados are fresh and halved. The bread is whole wheat and toasted.

Figure 7: More comparisons of different caption form and corresponding performances.

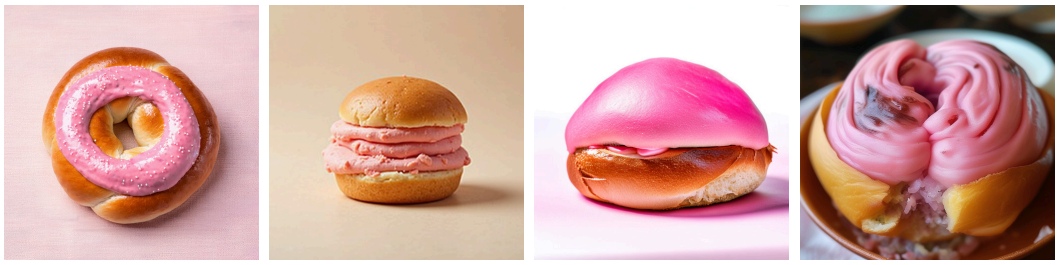
J More Visual Comparison Across Different Models

In this section, we present more qualitative comparisons across different models. Specifically, we employ our large-scale model DiT-ST Large and compare it with the baseline MM-DiT 8B-E under identical prompt conditions. As shown in Figure 8, DiT-ST Large produces more compositionally coherent and semantically aligned results, especially in complex scenes with diverse attributes.

Figure 8: Comparisons between MM-DiT 8B-E (left) and our DiT-ST Large (right)

We further compare DiT-ST Large with several state-of-the-art text-to-image generation models, including PixArt-α, Flux, and HunyuanDiT. As illustrated in Figure 10, our model demonstrates competitive or superior visual quality, with stronger object fidelity, spatial consistency, and semantic expressiveness.

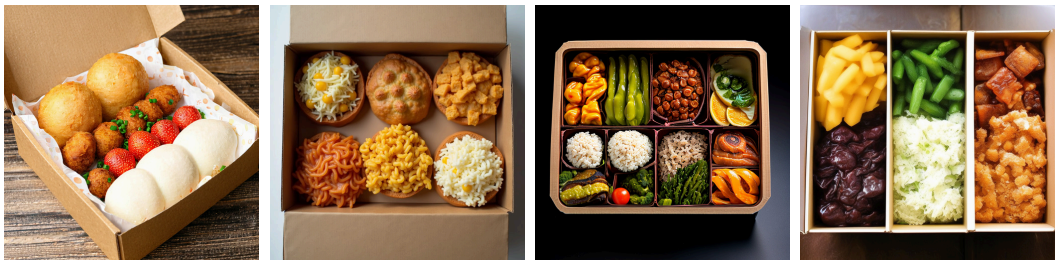
A person is skiing on snow, wearing gloves, a helmet, goggles, and a jacket, with a flag and trees in the background.



A bread roll with a golden, crispy exterior, coated in pink icing and sprinkled with white sugar pearls.

A group of teddy bears is wrapped in a blanket, with pillows underneath.

Under a shed with flowers, a man wearing a cap is working behind a table with piles of oranges and juice machine.



The box contains three types of food, with many fried items placed on a grease-absorbing paper lining.

A multi-colored wall clock is mounted on the wall, surrounded by a few sculptures, a label above reads "1106."

Figure 9: Comparisons among DiT-ST Large, Flux, PixArt-α and HunyuanDiT.

K More High Quality Images Generated by Our DiT-ST Large

Figure 10: High-quality 1024×1024 images generated by our DiT-ST Large

Besides the standard 1024×1024 images shown in Figure 10, we further generate multi-scale samples put in Figure 11, confirming that our DiT-ST Large maintains visual fidelity and semantic alignment across different resolutions and formats.

Figure 11: High-quality and multi-scale generation results by our DiT-ST Large

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification:The abstract and introduction clearly outline the core contributions of our work DiT-ST, including the motivation, methodological innovations, and performance gains.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We acknowledge the limitations of our work and will provide a dedicated discussion in the appendix.Rather than aiming to surpass all existing baselines, our contribution focuses on developing an effective and generalizable method that performs consistently across tasks and domains. We believe this design choice better serves long-term robustness and transferability.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: The paper includes the necessary theoretical results with all assumptions clearly stated and referenced in the respective sections. All proofs and lemmas are either presented within the main paper

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: To ensure full reproducibility, we release all necessary resources, including model weights, training and evaluation code, and preprocessed datasets. Detailed instructions for environment setup and running experiments are provided in GitHub repository.

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).

(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: We release both source code and necessary data to facilitate faithful reproduction of our results. The code repository includes training and inference scripts and environment configuration files. In addition, we provide access to all datasets used in our experiments on huggingface.

   Guidelines:

   - The answer NA means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: We describe all experimental settings in both the main paper and the appendix. Key implementation details such as data splits, model architecture, optimizer type, learning rate schedules, and hyperparameter configurations are summarized in the main text to support result interpretation.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [Yes]

Justification: We evaluate our method using commonly adopted standard metrics, including [e.g., FID, CLIPScore, or others], to ensure comparability with existing baselines. In addition to quantitative evaluation, we also provide visualizations of generated outputs to support qualitative assessment and highlight consistency across samples. These visual results, alongside statistical metrics, offer a comprehensive understanding of model behavior.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We specify in the main paper that all experiments were conducted on NVIDIA A100 GPUs.Each training run took approximately 12–24 GPU hours.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: We have carefully reviewed the NeurIPS Code of Ethics and confirm that our research fully conforms to its principles. Our work does not involve sensitive user data, human subjects, or harmful applications. All datasets used are either publicly available or properly licensed, and we ensure transparency, reproducibility, and responsible reporting throughout the paper.

Guidelines:

- The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: Our work introduces a novel and generalizable perspective for improving existing generative models, aiming to address foundational challenges in a model-agnostic manner. Rather than optimizing for narrow task-specific gains, our approach seeks to uncover transferable mechanisms that can be broadly applied across architectures and domains. This may positively influence future research toward more unified and interpretable generative modeling. While our method is not designed for deployment in sensitive or high-risk applications, we acknowledge the broader implications of advancing generation quality, and support responsible usage and release protocols.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [Yes]

    Justification: We acknowledge that high-quality generative models may pose risks of misuse, such as unauthorized image synthesis or disinformation. To mitigate such risks, we commit to releasing our pretrained models and code under a non-commercial research-only license. We also include usage guidelines outlining responsible applications, and plan to implement content safety filters to restrict harmful outputs. Furthermore, we will monitor the downstream use of our models and reserve the right to restrict access if misuse is identified.

    Guidelines:

- The answer NA means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We make proper attribution to all external assets used in our work. Specifically, we use the COCO 2017 dataset, which is publicly available under a Creative Commons Attribution 4.0 License (CC-BY 4.0), and we follow its terms of use. Our experiments are based on the Stable Diffusion v3 model provided by Stability AI, which is released under a custom non-commercial license; we cite the official source and follow the stated restrictions. We also build on open-source libraries from Hugging Face, such as Transformers and Diffusers, which are released under the Apache 2.0 license.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Justification: We do not introduce any new assets in this work. All experiments are conducted using publicly available datasets (e.g., COCO 2017) and existing pretrained models (e.g., Stable Diffusion v3). We ensure that these assets are properly cited and used in accordance with their respective licenses.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.

- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: Our work does not involve any form of crowdsourcing or research with human subjects. All experiments are conducted using publicly available datasets and pretrained models without collecting new human-provided annotations or feedback.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: This work does not involve research with human subjects or crowdsourcing. All experiments are conducted using publicly available datasets and pretrained models, without collecting new human annotations, feedback, or participation. Therefore, no IRB or equivalent ethical approval is required.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

    Answer: [Yes]

    Justification: We used the Qwen Plus API, a large language model, to assist in processing and structuring textual data used during pre-processing. This usage does not directly affect the core model architecture or training process and can be replaced by any other LLM models, but facilitates intermediate data organization and annotation required for our pipeline. We describe this usage and its scope in Appendix G to ensure transparency and reproducibility.

Guidelines:

- The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.