

# Attend to Not Attended: Structure-then-Detail Token Merging for Post-training DiT Acceleration

Haipeng Fang<sup>1,2</sup>, Sheng Tang<sup>1,2</sup>, Juan Cao<sup>1,2</sup>, Enshuo Zhang<sup>1,2</sup>, Fan Tang<sup>✉,1,2</sup>, Tong-ye Lee<sup>3</sup>

<sup>1</sup> Institute of Computing Technology, Chinese Academy of Sciences

<sup>2</sup> University of Chinese Academy of Sciences

<sup>3</sup> National Cheng-Kung University

## Abstract

Diffusion transformers have shown exceptional performance in visual generation but incur high computational costs. Token reduction techniques that compress models by sharing the denoising process among similar tokens have been introduced. However, existing approaches neglect the denoising priors of the diffusion models, leading to sub-optimal acceleration and diminished image quality. This study proposes a novel concept: **attend to** prune feature redundancies in areas **not attended** by the diffusion process. We analyze the location and degree of feature redundancies based on the structure-then-detail denoising priors. Subsequently, we introduce *SDTM*, a structure-then-detail token merging approach that dynamically compresses feature redundancies. Specifically, we design dynamic visual token merging, compression ratio adjusting, and prompt reweighting for different stages. Served in a post-training way, the proposed method can be integrated seamlessly into any DiT architecture. Extensive experiments across various backbones, schedulers, and datasets showcase the superiority of our method, for example, it achieves  $1.55\times$  acceleration with negligible impact on image quality. Project page: <https://github.com/ICTMCG/SDTM>.

## 1. Introduction

Diffusion transformers (DiTs) [33] are flourishing in image [6, 9, 11] and video [18, 30, 51] generation, and have been adopted as the fundamental model of Sora [3]. However, heavy computational redundancies slow down inference and drive the need for acceleration techniques.

Several sampler-based methods aim to optimize denoising steps through sampler optimization [27, 42] or distillation [31, 39], and model-based studies focus on pruning [4, 10], quantizing [12, 20], or caching [29, 45] architec-

✉Corresponding author: Fan Tang.

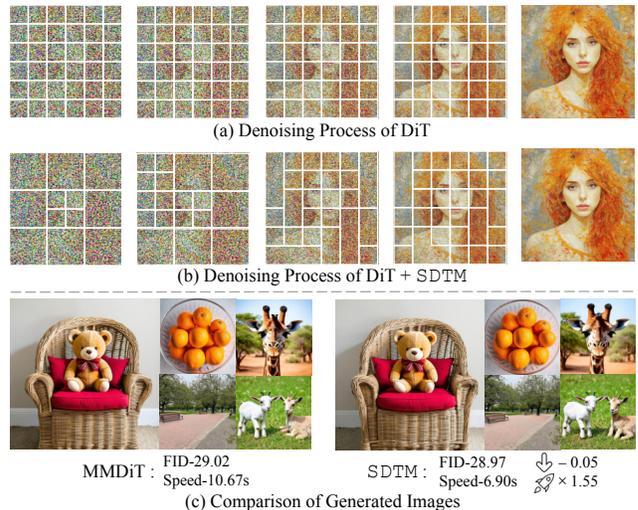


Figure 1. Illustration. *Upper*: Our SDTM represents a dynamic multi-resolution generation process by reducing feature redundancies in areas not attended by the denoising process. *Lower*: Compared to the baseline method, our approach achieves  $1.55\times$  acceleration with negligible impact on generation quality.

tural redundancy. Due to the inflexibility of these methods for diverse data and evolving requirements, various studies introduced token reduction [2, 22, 32, 37] to reduce feature redundancy. For instance, ToMeSD [1] and AT-EDM [44] compress similar or unimportant tokens, where DyDiT [50] and TokenCache [26] train a dynamic module to prune or cache unnecessary computation. However, these feature-based approaches often require additional fine-tuning or overlook the denoising priors, resulting in suboptimal acceleration, diminished image quality, and limited applicability.

In this paper, we propose a novel view: **attend to** prune feature redundancies in areas **not attended** by the diffusion process. Initially, we observe the evolution of low and high frequencies within DiT and confirm that diffusion transformers still adhere to the structure-then-detail denoising priors: they allocate less attention to less-structure tokens in

the early steps and to weak-detail tokens in the later steps. Subsequently, we hypothesize that these unattended tokens may be redundant, and we validate this hypothesis by tracking the location of feature redundancies. Furthermore, we find that the degree of feature redundancy in the early diffusion process is significantly greater than in the later stages.

Based on the above priors, we introduce SDTM, a novel structural-then-detail token merging approach that *dynamically* reduces token redundancies stepwise without requiring additional fine-tuning. For visual token merging, we develop a similarity-prioritized structural merging method, an inattentive-prioritized detail merging method, supplemented by time-wise ratio adjusting to alter the compression degree at different stages. From the perspective of prompt guidance, we design a time-wise prompt reweighting to optimize the guidance direction at different steps. In summary, the contributions of this paper are as follows:

- We analyze the location and degree of feature redundancies and introduce SDTM, a dynamic token compression approach that employs a structure-then-detail token merging strategy. It is finetuning-free and can be seamlessly integrated into any text-to-image DiT architecture.
- We design similarity-prioritized structural merging and inattentive-prioritized detail merging methods for different stages, supplemented by time-wise ratio adjusting and prompt reweighting to compress feature redundancies.
- Quantitative and qualitative experiments across multiple backbones, schedulers, and datasets demonstrate our method’s superiority. For example, SDTM achieves  $1.55\times$  acceleration with negligible impact on generation quality.

## 2. Related Work

### 2.1. Diffusion Transformers

Diffusion models (DMs) [8, 15, 43] transform noise into complex data distributions via reversible Markov processes. Early U-Net based DMs achieved remarkable results in image [34, 38, 48] and video generation [5, 40, 52]. Currently, diffusion transformers (DiTs) [33] are gaining prominence due to their scalability. Building on DiTs, image generation models such as PixArt- $\alpha$  [6], MMDiT [9], and FLUX [11] have been introduced. Furthermore, DiTs have been recognized as a fundamental component in the Sora [3], leading to the development of video generation models [18, 30, 51]. However, significant computational redundancies of diffusion sampling process limit its widespread application.

### 2.2. Efficient Diffusion Models

Numerous efforts have been proposed to compress the redundancies in the denoising process via samplers, architectures, and feature computation, it can be categorized as:

*Sampler-based:* Various approaches focus on optimizing samplers; for instance, DDIM [42] offers a non-Markovian

variant of the diffusion process, DPMSolver [27] applies a numerical solver to the differential equations and Rectified Flow [25] optimizes distribution transport in ODE models. Additionally, several progressive distillation techniques [17, 31, 39] attempt to distill the sampler into fewer steps.

*Model-based:* Some studies focus on reducing architecture redundancy: methods like Diff-pruning [4, 10, 16, 21] advocate pruning unimportant weights; quantization methods represented by Q-diffusion [12, 20, 41, 49] aim to quantize redundant modules. Furthermore, some caching mechanisms [7, 19, 29, 45] enhance efficiency by caching and reusing module outputs across adjacent denoising steps.

*Feature-based:* Given the dependency of sampler-based and model-based methods on fixed strategies, they struggle with varying data patterns and compression demands. Therefore, some approaches explore feature compression techniques. For instance, within the U-Net architecture, ToMeSD [1] and AT-EDM [44] advocate for compressing similar or unimportant tokens, whereas in DiT architectures, DyDiT [50] and TokenCache [26] train a dynamic token module to prune or cache unnecessary computation.

In contrast to other feature-based methods that utilize relatively *static* reduction strategies or require fine-tuning, our SDTM is specifically designed to perform *dynamic* post-training token reduction. This approach is based on thoroughly analyzing the location and degree of feature redundancies at different stages of diffusion process. Therefore, SDTM is more efficient compared to the above frameworks.

## 3. Preliminary and Motivation

Our approach is driven by a straightforward idea: *attend to* prune feature redundancies in areas *not attended* by the diffusion process. We assume that tokens unattended by the diffusion process might be redundant, and raise two following questions. Experiments were conducted on MMDiT [9] with 50-step RF schedule [25] based on 10k samples.

*Which ones are not attended by denoising process?* A well-known prior of the denoising process is that it includes a structure stage for planning visual semantics, followed by a detail stage that enhances the visual fidelity [27, 35, 46]. From this perspective, we aim to verify whether this prior still holds in DiT. Specifically, we analyzed the evolution of low-frequency and high-frequency components of the estimated noise using DWT. As illustrated in Figs. 2 (a) and (b), DiT’s architecture and scheduler have stabilized the frequency evolution, which fluctuated significantly in DDPM noted at [35]. The low-frequency bands show sharp changes in the first 40% steps, with the high-frequency bands undergoing alterations in the subsequent 60%. This pattern confirms that DiT maintains the structure-then-detail denoising process, highlighting that *less-structure tokens in early steps and weak-detail tokens in later steps are the unattended ones* which may be the redundant feature.

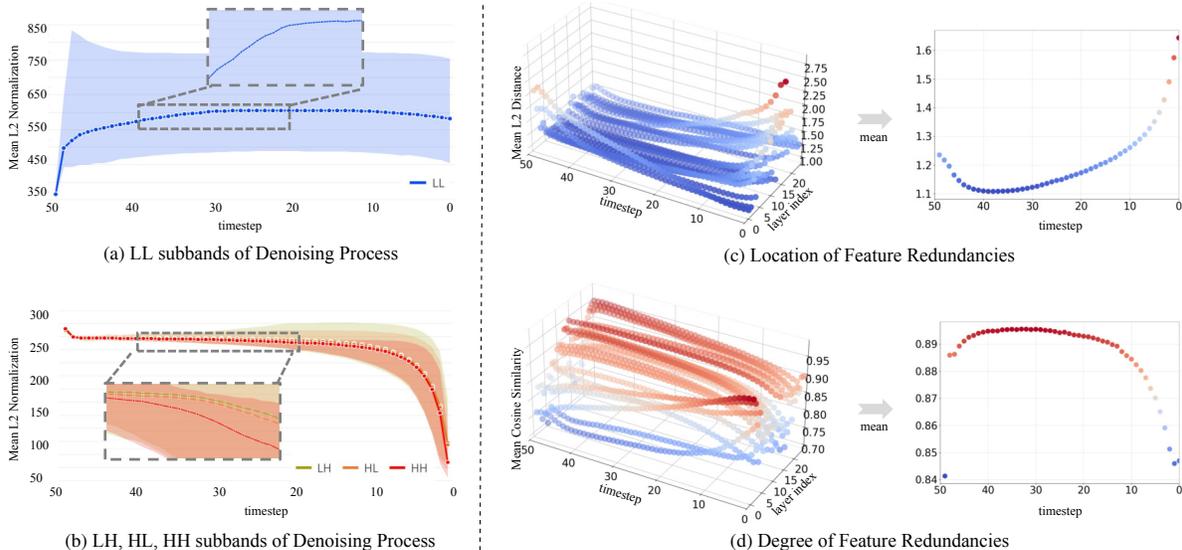


Figure 2. Preliminaries. *Left*: Evolution of Denoising Process: The L2 norm of (a) LL, (b) LH, HL, and HH subbands of estimated noise during the DiT denoising process. *Right*: Evolution of Feature Redundancies: (c) Location and (d) Degree evolution of token redundancies across DiT different steps and layers. Experiments were conducted on MMDiT [9] with 50-step RF schedule [25] based on 10k samples.

**How to find these unattended ones?** Firstly, we locate these unattended tokens and assess their redundancy. Specifically, we calculated the mean L2 distance and cosine similarity between the most similar pairs. Higher similarity indicates that they can substitute for each other which represents more redundancies [1], while the distance pinpoints their location. As illustrated in Fig. 2 (c), the distance decreases slightly and then increases progressively. Since higher similarity at lower distances implies less local structure, this identifies the initial locations of unattended tokens. Meanwhile, decreasing similarity indicates increasing detail diversity, with unattended tokens distributed among less-detail global areas. Therefore, the unattended tokens, which represent feature redundancies, initially cluster locally and gradually spread globally. Secondly, we analyzed the degree evolution of cosine similarity in Fig. 2 (d). Besides the initial low similarity among tokens immediately after the noise initialization, which quickly escalates after one step, the token similarity progressively decreases as timesteps, implying that token redundancy diminishes over time. Therefore, there are more unattended, redundant tokens in the early steps and fewer in the later steps.

## 4. Methodology

As shown in Fig. 3, we introduce the SDTM framework, a structure-then-detail token merging approach that facilitates dynamic token compression to reduce feature redundancy. Based on the analysis in Sec. 3, we perform DiT acceleration across two stages. At different stages, we adopt various visual token merging (shown in Fig. 4) and prompt token reweighting methods, detailed in Sec. 4.1 and Sec. 4.2.

### 4.1. Visual Token Merging

To address feature redundancies vary in form and location, we design a similarity-prioritized structure merging (SSM) to address local redundancies in structure stage and an inattentive-prioritized detail merging (IDM) to reduce global redundancies in detail stage. Furthermore, considering the higher redundancies in early steps and lower in later steps, we develop compression ratio adjusting (CRA) that dynamically optimizes ratios or thresholds across timesteps.

#### 4.1.1. Similarity-prioritized Structure Merging

Based on the insights derived from Sec. 3, we reduce the weak-structure feature redundancy in local areas during the early stage. We developed similarity-prioritized structure merging and integrated them before the MHSA and MLP blocks. The SSM involves two processes: identifying high-redundancy tokens and merging them dynamically.

**Identifying.** In the initial stage, tokens with less structure are identified within local areas. The feature embedding is represented by  $\mathcal{X} \in \mathbb{R}^{N \times d}$ , where  $N = H \times W$  and  $H, W$  denote the height and width. Inspired by ALGM [32], we reshape  $\mathcal{X}$  into a grid  $\mathcal{X} \in \mathbb{R}^{m \times \frac{H}{m} \times m \times \frac{W}{m} \times d}$ , with  $m \times m$  indicates the window size. By grouping the tokens of each window, we define  $\mathcal{X} = \{w_1, \dots, w_k, \dots\}$ . Subsequently, we compute the average cosine similarity within each window  $w$  as the similarity priority score  $\mathcal{P}_w^{sim}$ :

$$\mathcal{P}_w^{sim} = \frac{\sum_{x_i, x_j \in w}^{i \neq j} \cos(x_i, x_j)}{m^2 \cdot (m^2 - 1)}. \quad (1)$$

Additionally, we observed that error accumulation tends to escalate when a token is continuously merged. To mitigate

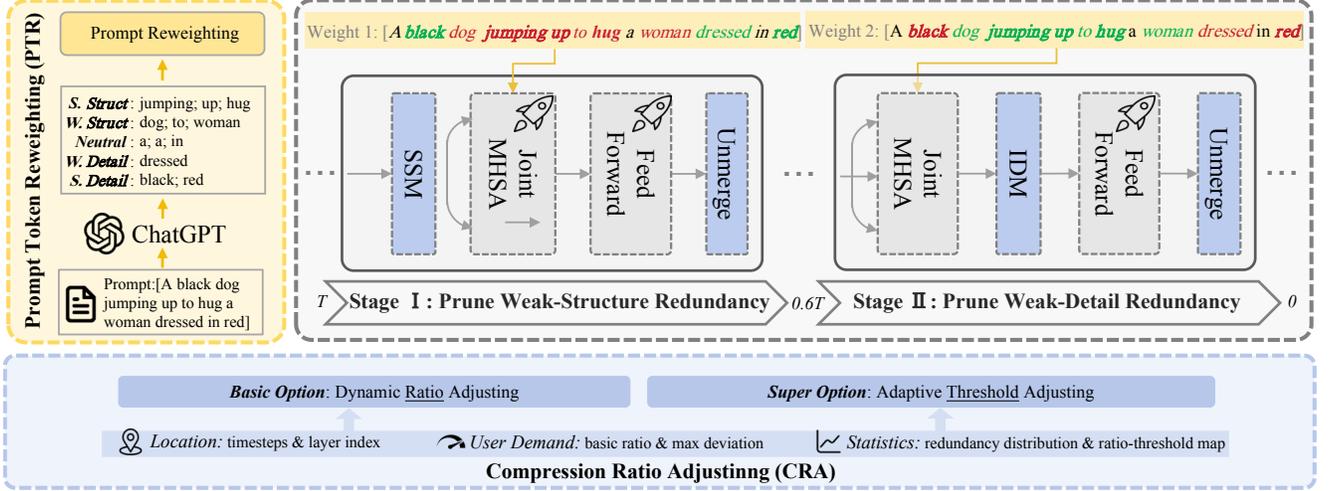


Figure 3. Overview. *Grey*: Our SDTM compresses weak-structure redundancies in the early stage and weak-detail redundancies in the later stage. *Blue*: Compression ratio adjusting (CRA) dynamically adjusts the ratio or threshold to control the pruning degree. *Yellow*: Prompt token reweighting (PTR) categorizes each prompt token into structure or detail groups, optimizing the denoising direction by reweighting attention map. Here, increase and decrease in red and green, while bold implies intensity.

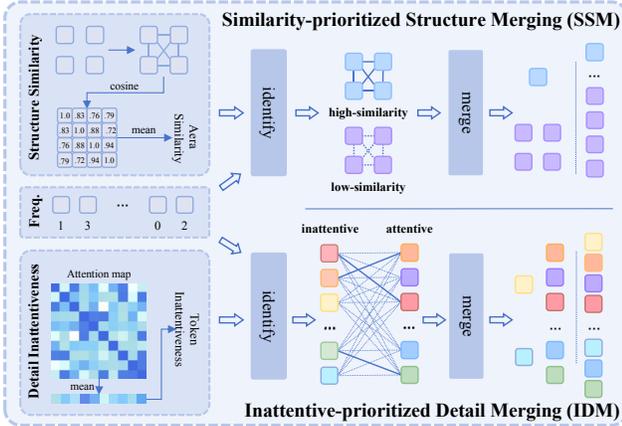


Figure 4. Visual Token Merging. By measuring structure similarity, unmerged frequency, and detail inattentiveness, SSM and IDM target different types of feature redundancies for reduction.

this, we prioritize tokens unmerged recently over those involved in continuous merging events. We monitor the time since each token’s last independent operation, represented as  $\mathcal{T}_x$ , and calculate  $\mathcal{P}_x^{fre} = \frac{\mathcal{T}_x}{\mu(\mathcal{T})}$  for each token. We calculate frequency priority score  $\mathcal{P}_w^{fre}$  for window  $w$ :

$$\mathcal{P}_w^{fre} = \frac{\sum_{x_i \in w} \mathcal{P}_{x_i}^{fre}}{m^2}. \quad (2)$$

Finally, we compute the total priority score  $\mathcal{P}_w = \mathcal{P}_w^{sim} + \alpha_s \mathcal{P}_w^{fre}$ , where  $\alpha_s$  is a scaling factor. We sort the windows in descending order to prioritize the redundancies.

**Merging.** Our merging strategy can be controlled using either a ratio  $\rho$  or a threshold  $\theta$ . We select either the top  $\rho$

redundant windows or windows where  $\mathcal{P}_w > \theta$  for merging. Both  $\rho$  and  $\theta$  are dynamically adjusted throughout the timesteps, as detailed in Sec. 4.1.3. Subsequently, we average the tokens within each selected window and retain the tokens from unselected windows to construct a new feature embedding  $\mathcal{X}' \in \mathbb{R}^{N' \times d}$ . In this process,  $N'$  is reduced compared to  $N$ , thereby decreasing the computational cost on the subsequent MSHA and MLP blocks.

#### 4.1.2. Inattentive-prioritized Detail Merging

Following the analyses in Sec. 3, we reduce weak-detail feature redundancy in later stages of the denoising process. We developed inattentive-prioritized detail merging and incorporated them before the MLP blocks. We have ceased accelerating the MSHA module, as its role in facilitating global information interactions is crucial, and its computational demand is significantly lower than that of MLP. Similarly, the IDM includes identifying and merging processes.

**Identifying.** We assume that tokens with minimal impact on others are information-sparse and thus weak-detail. Utilizing the attention map, which effectively quantifies the relationships between tokens, we identify the inattentive tokens. For attention map  $\mathcal{A} \in \mathbb{R}^{N \times N}$ ,  $\mathcal{A}(x_i, x_j)$  quantifies the influence of the  $j$ -th token on the  $i$ -th token, a notion widely recognized in [23, 37]. We then compute the inattentive priority score  $\mathcal{P}_x^{ina}$  for each token as follows:

$$\mathcal{P}_x^{ina} = -\frac{\sum_{j \in 1 \dots N} \mathcal{A}(x, x_j)}{N}. \quad (3)$$

Subsequently, we calculate  $\mathcal{P}_x^{fre}$  for each token and its overall priority score  $\mathcal{P}_x = \mathcal{P}_x^{ina} + \alpha_d \mathcal{P}_x^{fre}$ , where  $\alpha_d$  is

a scaling factor. We then sort the tokens in descending order to effectively prioritize the feature redundancies.

**Merging.** Based on the priority score, we categorize tokens into inattentive and attentive groups, as shown in Fig. 4. We calculate the cosine similarity between these groups and identify the maximum cosine similarity  $\mathcal{S}_i$  for each token in inattentive group. Depending on the merging ratio  $\rho$  or threshold  $\theta$ , we select either the top  $\rho$  redundant tokens or those which  $\mathcal{S}_i > \theta$  for merging. It is essential that  $\mathcal{P}$  determines the merging priority, whereas the merging process relies on  $\mathcal{S}$ . Considering  $x_i$  and  $x_a$  as an example, given the differing information content between inattentive and attentive groups, the merged  $x'_a$  can be formulated as:

$$x'_a = \text{softmax}([1 - \mathcal{P}_{x_i}^{ina}, 1 - \mathcal{P}_{x_a}^{ina}]) \cdot [x_i, x_a]. \quad (4)$$

#### 4.1.3. Compression Ratio Adjusting

The variability in the degree of feature redundancies across timesteps is evident, as shown in Fig. 2. With increasing timesteps, there is a general decline in the degree of feature redundancies. In addition, specific steps (the first) and specific layers (the first four) deviate from this trend. In practical scenarios, users often have specific compression requirements, which typically include a basic ratio  $\rho$  and a maximum deviation  $d$ . To meet these demands, we propose two strategies: dynamically adjusting the compression ratio and adaptively adjusting the compression threshold.

**Dynamic Ratio.** This method offers a straightforward approach for dynamically adjusting the merging ratio. It modifies the ratio to follow a cosine decay of  $[0, \frac{\pi}{2}]$  from  $\rho + d$  to  $\rho - d$  across various steps, in alignment with the cosine-shaped curve of feature redundancies illustrated in Fig. 2. For specific steps and layers, the ratio is set directly to the minimum  $\rho - d$ . We adopt this strategy in our baseline SDTM model because it provides a close approximation of the optimal ratio without the need for complex computations.

**Adaptive Threshold.** Due to the varying complexity of generating different images, employing a constant merging ratio may result in suboptimal acceleration for simpler images and compromised quality for more complex ones. To address this issue, we design an adaptive method to automatically adjust the threshold. We initially sampled a small image batch to assess the similarity across steps and layers to create a distribution  $S$ . Then, we constructed a ratio-threshold mapping table  $M$  by documenting the average threshold at 1% intervals of the merging ratio. The comprehensive process is outlined in Algorithm 1. This approach has been integrated into our enhanced SDTM\*, better suited for industrial-scale inference scenarios where neither suboptimal acceleration nor quality degradation is acceptable.

#### 4.1.4. Token Unmerging

Image generation is a feature-intensive task that necessitates the complete feature map. Using the example of  $x_i$  and

---

#### Algorithm 1 Adaptive Threshold Adjusting

---

**Require:** similarity distribution  $S$ , basic ratio  $r$ , max deviation  $d$ , ratio-threshold map  $M$

**Input:** timestep & layer sequence  $\{(1, 1), \dots, (t, l)\}$

**Output:** adaptive threshold  $\theta = \{\theta_{(1,1)}, \dots, \theta_{(t,l)}\}$

- 1: Scale similarity distribution  $S$  to range  $[-1, 1]$ .
  - 2: **for each**  $t$  **do**
  - 3:   **for each**  $l$  **do**
  - 4:     Compute  $S_{t,l}$  for current  $t$  and  $l$
  - 5:      $\rho_{(t,l)} \leftarrow r + d \cdot S_{(t,l)}$
  - 6:      $\theta_{(t,l)} \leftarrow M(t, l, \rho_{(t,l)})$
  - 7:   **end for**
  - 8: **end for**
- 

$x_a$ , we need to use  $x'_a$  to obtain new  $x''_i$  and  $x''_a$ . In related work such as ToMeSD [1] and AT-EDM [44], the strategy of similarity-based token reuse has been utilized, directly substituting  $x'_a$  for  $x''_i$  and  $x''_a$ . However, this substitution can introduce replacement errors, particularly when employing a higher pruning ratio. Therefore, we initially employ similarity-based token reuse to restore  $x''(t)$  at the current timestep, followed by a weighted combination with the previous timestep’s  $x''(t-1)$ . In this method,  $x''(t-1)$  preserves the independence of individual token features, while  $x''(t)$  integrates the new timestep’s denoising features.

## 4.2. Prompt Token Reweighting

The image generation process evolves from capturing the overall structure to refining intricate details, emphasizing the importance of the influence of prompt tokens at various stages [13]. Moreover, as visual tokens become substantially compressed, the direction of prompt guidance grows increasingly vital. We developed prompt token reweighting to optimize the guiding direction across timesteps. We define the “prompt” as the image description, whereas “instruction” is the query to ChatGPT. As shown in Fig. 3, given a prompt  $\mathbb{P}$ , we guide ChatGPT using the following interaction to categorize each prompt token  $\mathbb{P}_k$ .

**Instruction:** Categorize Prompt Token

**System Instruction:** Suppose you are a data scientist. You will be provided with an [prompt]. You should categorize each prompt token into five categories: Strong Structure, Weak Structure, Neutral, Weak Detail, and Strong Detail. If a token contains both structure and detail, weight them for decision.

**Context Instruction:** [prompt  $\mathbb{P}$ ]

We multiply the attention values by an optimized range  $[\alpha_p, \frac{1}{\alpha_p}]$ . Take examples from Fig. 3, the adjustment for weakly structured prompt token “dog” is  $[\frac{\alpha_p}{2}, \frac{2}{\alpha_p}]$ , while for strongly detailed prompt token “black” is  $[\frac{1}{\alpha_p}, \alpha_p]$ .

Method	Finetune	MACs(T) ↓	Latency(s) ↓	Speed ↑	COCO2017		PartiPrompts
					FID ↓	CLIP ↑	CLIP ↑
<b>SD3 Medium [9]</b>		6.01	10.67	1.00×	29.02	0.3267	0.3279
ToMeSD - a		4.27	8.08	1.32×	45.28	0.3102	0.3108
AT-EDM - a		4.23	8.14	1.31×	34.72	0.3195	0.3219
Ours-SDTM - a		4.20	8.20	1.30×	28.73	0.3235	0.3248
Ours-SDTM* - a		<b>4.13</b>	<b>8.02</b>	<b>1.33×</b>	<b>28.57</b>	<b>0.3249</b>	<b>0.3261</b>
ToMeSD - b		3.81	7.11	1.50×	75.44	0.2780	0.2816
AT-EDM - b		3.78	7.02	1.52×	43.92	0.3089	0.3125
TokenCache - b	✓	3.72	6.97	1.53×	28.83	0.3208	0.3226
DyDiT - b	✓	3.74	<b>6.87</b>	<b>1.55×</b>	<b>28.47</b>	0.3213	0.3227
Ours-SDTM - b		3.66	7.07	1.51×	29.60	0.3224	0.3231
Ours-SDTM* - b		<b>3.62</b>	6.90	<b>1.55×</b>	28.97	<b>0.3237</b>	<b>0.3252</b>

Table 1. **Quantitative comparison** on MS-COCO2017 and PartiPrompts with Stable Diffusion 3 medium and 50 steps rectified flow by default. For configurations a and b, we adjust the compression ratios of various methods to reach approximate speeds of 1.3× and 1.5×.

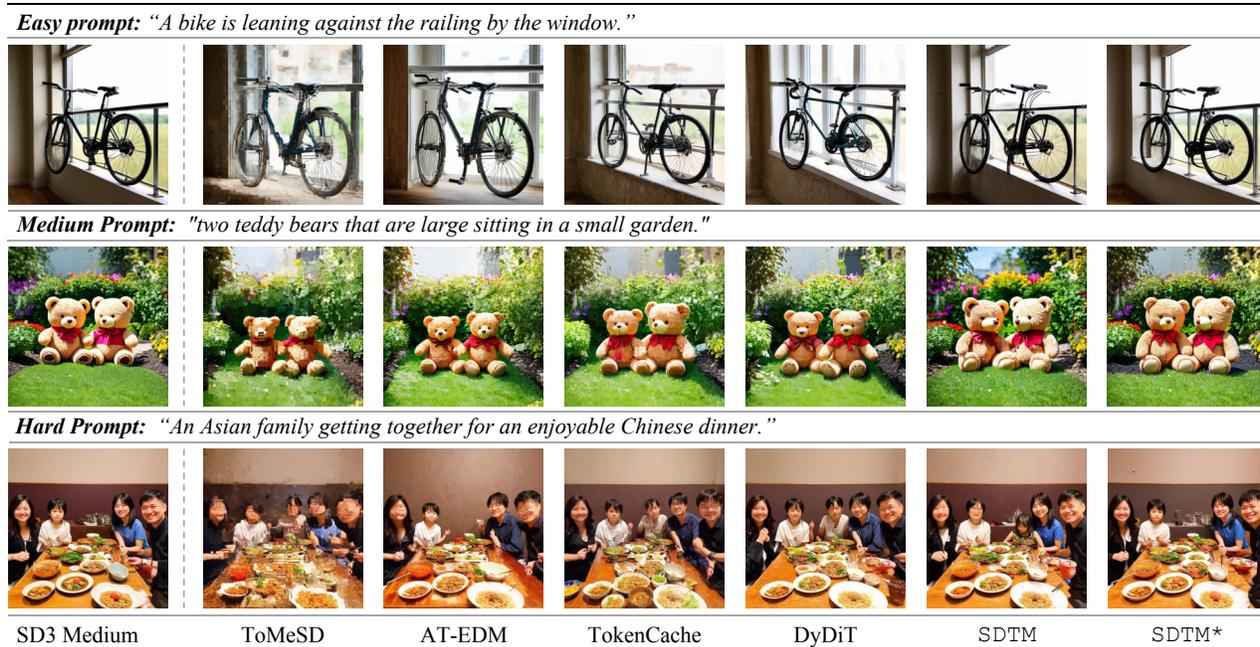


Figure 5. **Qualitative comparison** on COCO2017 and PartiPrompts under varying data complexities. For ToMeSD and AT-EDM, we use versions with approximately 1.3× acceleration, while others use approximately 1.5× versions. Best viewed when zoomed in.

## 5. Experiment

### 5.1. Experiments Settings

**Implementation details.** Our method can be seamlessly integrated into any text-to-image DiT architecture to facilitate post-training acceleration. It is available in two versions: SDTM is a straightforward implementation, and SDTM\* adaptively adjusts the compression threshold based on the image’s complexity. Unless specified otherwise, we set the initial  $T - 0.6T$  as the structure stage and the remaining  $0.6T - 0$  as the detail stage, using the hyperparameters basic ratio  $\rho$  and maximum deviation  $d$  to 0.5 and 0.2.

**Evaluations.** We conduct extensive quantitative and qual-

itative experiments on various model configurations, including SD3 Medium, SD3.5 Large and SD3.5 Large Turbo [9], utilizing different schedulers such as Restricted Flow [37], DPM-Solver++ [28] across varying denoising steps (e.g. 50, 28, 20, 15). Following the protocol in AT-EDM [44], our experiments were executed primarily on the COCO2017 validation set [24] and PartiPrompt [47] at an image resolution of  $1024 \times 1024$ . We evaluated performance using MACs and Latency, alongside FID [14] and CLIP scores [36] for image quality assessment. Latency was calculated by the average time required to generate 5000 images on COCO2017 validation. All experiments were performed using 4 NVIDIA A100 40G GPUs.

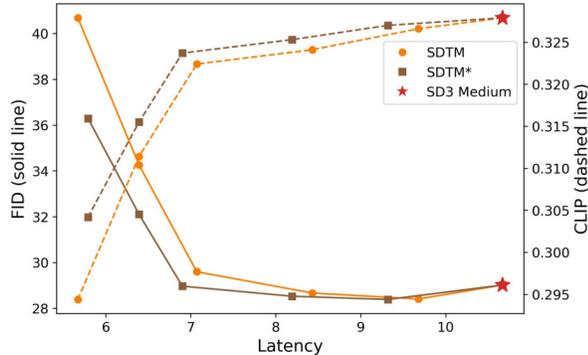


Figure 6. The trade-off of Latency vs. FID and CLIP. We obtain SDTM and SDTM\* with different latency by adjusting the ratio.

## 5.2. Comparisons with SOTAs

Table 1 compares our method with existing token-wise feature redundancy compression techniques. In configuration a, ToMeSD and AT-EDM show noticeable quality degradation, whereas our SDTM and SDTM\* slightly improve image quality with reductions in FID by 0.29 and 0.45. In configuration b, as ToMeSD and AT-EDM exhibit significant quality declines, we introduce comparisons with finetuning-based TokenCache and DyDiT. Our approach achieves comparable image quality without the need for fine-tuning, benefiting from our detailed analysis and targeted reduction of feature redundancy at various stages. Our method achieves optimal CLIP, leveraging the prompt reweighting strategy that maintains directional guidance with fewer compressed tokens, which are detailed in ablation. Qualitative comparisons in Fig. 5 further support our findings. Moreover, SDTM\* adaptively adjusts the merging threshold for samples of varying difficulty, allocating limited computational resources more effectively and surpassing other methods in image quality and alignment with the original images.

## 5.3. Comparisons with baselines

We further integrate SDTM and SDTM\* into more baselines to evaluate their compatibility. As indicated in Fig. 6 and Fig. 7, we evaluate the SD3 Medium integrated with our method across a range of basic ratios  $\rho$  with the optimal deviation  $d$  set at 0.2. Results show that image quality remains stable with  $1.55\times$  acceleration (config b). However, further increasing the acceleration ratio progressively degrades image quality, confirming that config b is the most advantageous balance. This phenomenon is attributed to the distribution of redundancy in the images: compression ratios that are too low fail to harness token merging techniques fully. In contrast, excessively high ratios cause a rapid accumulation of merging errors. Additionally, as shown in Table 2, we extend SDTM and SDTM\* to SD3 Medium and SD3.5 Large to explore the impact of different denoising steps (28, 20, and 15) and various schedulers (RF, DPM-Solver++).



Figure 7. Visualization of results for graduated basic ratios.

Method	Scheduler	W-MACs(T) ↓	Latency(s) ↓	FID ↓
SD3 Medium	28-RF	168.4	6.29	28.74
+SDTM	28-RF	103.3	4.22	28.95
+SDTM*	28-RF	101.5	4.11	28.67
SD3 Medium	20-RF	120.3	4.68	28.86
+SDTM	20-RF	74.4	3.15	29.10
+SDTM*	20-RF	72.5	3.08	28.74
SD3 Medium	20-DPM	120.3	4.68	29.04
+SDTM	20-DPM	74.4	3.16	29.31
+SDTM*	20-DPM	72.9	3.12	29.08
SD3 Medium	15-RF	90.2	3.61	29.36
+SDTM	15-RF	56.3	2.45	29.82
+SDTM*	15-RF	54.4	2.43	29.51
SD3.5 Large	28-RF	668.4	18.54	25.91
+SDTM	28-RF	411.2	12.10	25.85
+SDTM*	28-RF	402.6	11.94	25.72

Table 2. Comparison of our SDTM and SDTM approaches with SD3 Medium and SD3 Large across different schedulers. Here, “W-MACs” represent the total computation across all steps.\*

Our method shows high adaptability across diverse baselines and schedulers. *More comparisons are available in the supplementary material.*

## 5.4. Ablation studies

We conduct ablation studies on the primary components of our method as outlined below. *More detailed experiments and analyses are available in the supplementary material.*

**Effect of token merging strategies.** To validate the effectiveness of our proposed structure-then-detail token merging strategies, we performed ablation studies using different combinations of merging strategies, as illustrated in Fig. 8. The results indicate that employing SSM in the early stages and IDM in the later stages yields optimal performance, evidenced by an FID score of 29.60. In contrast, the IDM-then-SSM sequence recorded the poorest results, with an FID score of 34.52. These results align with our previous analysis in Sec. 3, which posits that feature redundancy pre-



Figure 8. Ablation study of different token merging combinations at various stages. SSM in early stages and IDM in later stages performs the best, while the reverse config performs the worst.

Method	deviation	decline	FID	CLIP
SDTM	0.1	cosine	30.53	0.3216
	0.2	linear	31.64	0.3202
	0.2	cosine	29.60	0.3224
	0.3	cosine	31.81	0.3198
	0.4	cosine	36.03	0.3187

Table 3. Ablation of deviation values and ratio decline strategies. When the deviation of 0.2 and decline of cosine, the best results are achieved. We mark the optimal trade-off setting by <sup>†</sup>.

dominantly occurs among locally less-structure tokens in early stages and globally less-detail tokens in later stages.

**Effect of compression ratio adjusting.** Due to varying degrees of feature redundancy across denoising stages, we developed dynamic ratio adjusting and adaptive threshold adjusting strategies. Notably, the adaptive threshold adjustment mechanism distinguishes SDTM\* from SDTM, as demonstrated by its ability to dynamically optimize merging ratios for samples with diverse complexities (as shown in Fig. 6 and Table 2). We conducted ablation on the deviation value  $d$  and ratio decline strategies within the dynamic ratio adjusting framework in Table 3. Our results indicate that a deviation of  $d = 0.2$  achieves the optimal trade-off, while cosine decay yields superior performance compared to linear decay. Furthermore, Fig. 9 illustrates the progressive merging process within SSM and IDM merging strategies throughout the generation process.

**Effect of prompt token reweighting.** PTR optimizes the guidance direction at various stages. Fig. 10 shows that employing PTR slightly enhances the FID and significantly improves CLIP. We address a prevalent issue identified in ToMeSD and AT-EDM: token compression undermines CLIP. We attribute this to reduced attention to images due to the fewer tokens caused by compression. Fig. 10 also shows that the absence of PTR leads to color, structural errors, and semantic misunderstandings. Therefore, while PTR does not directly reduce computation costs, it is crucial to maintain alignment with prompts.

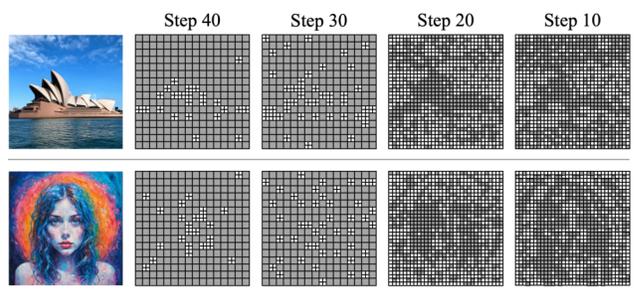


Figure 9. Visualization of merged tokens selected. White masks represent independent sampling, while gray masks represent merging. In the later stage, only inattentive merged tokens are grayed.



Figure 10. Ablation of prompt token reweighting (PTR). From left to right, the absence of PTR leads to minor color, structural errors, and severe semantic misunderstandings.

## 6. Conclusion

In this paper, we conduct a detailed analysis of the location and degree of feature redundancies and design a novel approach to accelerate DiTs by targeting redundancies in areas overlooked by the denoising process. Our innovative SDTM method dynamically addresses less-structure and less-detail redundancy throughout the generation process. It can be integrated seamlessly into any existing DiT architecture, accelerating generation without additional fine-tuning. We conducted extensive quantitative and qualitative experiments to demonstrate the effectiveness of our method across various architectures and schedulers. **Limitation:** Due to the inherent trade-offs of compression ratios in token merging, greater compression demands necessitate the integration of other acceleration techniques, such as distillation.

**Acknowledgements.** This work was partly supported by the Beijing Science and Technology Plan Project under No. Z231100005923033, Beijing Natural Science Foundation under No. L221013, and National Science and Technology Council under Grant 113-2221-E-006-161-MY3, Taiwan.

## References

- [1] Daniel Bolya and Judy Hoffman. Token merging for fast stable diffusion. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 4599–4603, 2023. 1, 2, 3, 5
- [2] Daniel Bolya, Cheng-Yang Fu, Xiaoliang Dai, Peizhao Zhang, Christoph Feichtenhofer, and Judy Hoffman. Token merging: Your vit but faster. In ICLR, 2023. 1
- [3] Tim Brooks, Bill Peebles, Connor Holmes, Will DePue, Yufei Guo, Li Jing, David Schnurr, Joe Taylor, Troy Luhman, Eric Luhman, Clarence Ng, Ricky Wang, and Aditya Ramesh. Video generation models as world simulators. 2024. 1, 2
- [4] Thibault Castells, Hyoungh-Kyu Song, Bo-Kyeong Kim, and Shinkook Choi. Ld-pruner: Efficient pruning of latent diffusion models using task-agnostic insights. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 821–830, 2024. 1, 2
- [5] Haoxin Chen, Menghan Xia, Yingqing He, Yong Zhang, Xiaodong Cun, Shaoshu Yang, Jinbo Xing, Yaofang Liu, Qifeng Chen, Xintao Wang, et al. Videocrafter1: Open diffusion models for high-quality video generation. arXiv preprint arXiv:2310.19512, 2023. 2
- [6] Junsong Chen, Jincheng Yu, Chongjian Ge, Lewei Yao, Enze Xie, Yue Wu, Zhongdao Wang, James Kwok, Ping Luo, Huchuan Lu, et al. Pixart- $\alpha$ : Fast training of diffusion transformer for photorealistic text-to-image synthesis. arXiv preprint arXiv:2310.00426, 2023. 1, 2
- [7] Pengtao Chen, Mingzhu Shen, Peng Ye, Jianjian Cao, Chongjun Tu, Christos-Savvas Bouganis, Yiren Zhao, and Tao Chen. -dit: A training-free acceleration method tailored for diffusion transformers. arXiv preprint arXiv:2406.01125, 2024. 2
- [8] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. Advances in neural information processing systems, 34:8780–8794, 2021. 2
- [9] 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. 1, 2, 3, 6
- [10] Gongfan Fang, Xinyin Ma, and Xinchao Wang. Structural pruning for diffusion models. Advances in neural information processing systems, 36, 2024. 1, 2
- [11] Black forest labs. Flux.1. <https://github.com/black-forest-labs/flux>, 2024. 1, 2
- [12] Yefei He, Luping Liu, Jing Liu, Weijia Wu, Hong Zhou, and Bohan Zhuang. Ptdq: Accurate post-training quantization for diffusion models. Advances in Neural Information Processing Systems, 36, 2024. 1, 2
- [13] 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. 5
- [14] 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. 2017. 6
- [15] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. pages 6840–6851, 2020. 2
- [16] Bo-Kyeong Kim, Hyoungh-Kyu Song, Thibault Castells, and Shinkook Choi. Bk-sdm: A lightweight, fast, and cheap version of stable diffusion. arXiv preprint arXiv:2305.15798, 2023. 2
- [17] Sanghwan Kim, Hao Tang, and Fisher Yu. Distilling ode solvers of diffusion models into smaller steps. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 9410–9419, 2024. 2
- [18] PKU-Yuan Lab and Tuzhan AI etc. Open-sora-plan. <https://doi.org/10.5281/zenodo.10948109>, 2024. 1, 2
- [19] Senmao Li, Taihang Hu, Fahad Shahbaz Khan, Linxuan Li, Shiqi Yang, Yaxing Wang, Ming-Ming Cheng, and Jian Yang. Faster diffusion: Rethinking the role of unet encoder in diffusion models. arXiv e-prints, pages arXiv–2312, 2023. 2
- [20] Xiuyu Li, Yijiang Liu, Long Lian, Huanrui Yang, Zhen Dong, Daniel Kang, Shanghang Zhang, and Kurt Keutzer. Q-diffusion: Quantizing diffusion models. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pages 17535–17545, 2023. 1, 2
- [21] Yanyu Li, Huan Wang, Qing Jin, Ju Hu, Pavlo Chemerys, Yun Fu, Yanzhi Wang, Sergey Tulyakov, and Jian Ren. Snapfusion: Text-to-image diffusion model on mobile devices within two seconds. Advances in Neural Information Processing Systems, 36, 2024. 2
- [22] Youwei Liang, Chongjian Ge, Zhan Tong, Yibing Song, Jue Wang, and Pengtao Xie. Not all patches are what you need: Expediting vision transformers via token reorganizations. In ICLR, 2022. 1
- [23] Youwei Liang, Chongjian Ge, Zhan Tong, Yibing Song, Jue Wang, and Pengtao Xie. Evit: Expediting vision transformers via token reorganizations. In ICLR, 2022. 4
- [24] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: Common objects in context. In Proceedings of the European Conference on Computer Vision, pages 740–755. Springer, 2014. 6
- [25] Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. arXiv preprint arXiv:2209.03003, 2022. 2, 3
- [26] Jinming Lou, Wenyang Luo, Yufan Liu, Bing Li, Xinmiao Ding, Weiming Hu, Jiajiong Cao, Yuming Li, and Chengang Ma. Token caching for diffusion transformer acceleration. arXiv preprint arXiv:2409.18523, 2024. 1, 2
- [27] Cheng Lu, Yuhao Zhou, Fan Bao, Jianfei Chen, Chongxuan Li, and Jun Zhu. Dpm-solver: A fast ode solver for diffusion probabilistic model sampling in around 10 steps. Advances in Neural Information Processing Systems, 35:5775–5787, 2022. 1, 2
- [28] Cheng Lu, Yuhao Zhou, Fan Bao, Jianfei Chen, Chongxuan Li, and Jun Zhu. Dpm-solver++: Fast solver for guided sampling of diffusion probabilistic models. arXiv preprint arXiv:2211.01095, 2022. 6

- [29] Xinyin Ma, Gongfan Fang, and Xinchao Wang. Deepcache: Accelerating diffusion models for free. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 15762–15772, 2024. 1, 2
- [30] Xin Ma, Yaohui Wang, Gengyun Jia, Xinyuan Chen, Ziwei Liu, Yuan-Fang Li, Cunjian Chen, and Yu Qiao. Latte: Latent diffusion transformer for video generation. arXiv preprint arXiv:2401.03048, 2024. 1, 2
- [31] Chenlin Meng, Robin Rombach, Ruiqi Gao, Diederik Kingma, Stefano Ermon, Jonathan Ho, and Tim Salimans. On distillation of guided diffusion models. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 14297–14306, 2023. 1, 2
- [32] Narges Norouzi, Svetlana Orlova, Daan de Geus, and Gijs Dubbelman. Algm: Adaptive local-then-global token merging for efficient semantic segmentation with plain vision transformers. In CVPR, pages 15773–15782, 2024. 1, 3
- [33] 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. 1, 2
- [34] Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. Sdxl: Improving latent diffusion models for high-resolution image synthesis. arXiv preprint arXiv:2307.01952, 2023. 2
- [35] Yurui Qian, Qi Cai, Yingwei Pan, Yehao Li, Ting Yao, Qibin Sun, and Tao Mei. Boosting diffusion models with moving average sampling in frequency domain. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 8911–8920, 2024. 2
- [36] 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. 6
- [37] Yongming Rao, Wenliang Zhao, Benlin Liu, Jiwen Lu, Jie Zhou, and Cho-Jui Hsieh. Dynamicvit: Efficient vision transformers with dynamic token sparsification. In NeurIPS, 2021. 1, 4, 6
- [38] 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. 2
- [39] Tim Salimans and Jonathan Ho. Progressive distillation for fast sampling of diffusion models. arXiv preprint arXiv:2202.00512, 2022. 1, 2
- [40] Uriel Singer, Adam Polyak, Thomas Hayes, Xi Yin, Jie An, Songyang Zhang, Qiyuan Hu, Harry Yang, Oron Ashual, Oran Gafni, et al. Make-a-video: Text-to-video generation without text-video data. arXiv preprint arXiv:2209.14792, 2022. 2
- [41] Junhyuk So, Jungwon Lee, Daehyun Ahn, Hyungjun Kim, and Eunhyeok Park. Temporal dynamic quantization for diffusion models. Advances in Neural Information Processing Systems, 36, 2024. 2
- [42] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In International Conference on Learning Representations, 2020. 1, 2
- [43] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In International Conference on Learning Representations, 2020. 2
- [44] Hongjie Wang, Difan Liu, Yan Kang, Yijun Li, Zhe Lin, Niraj K Jha, and Yuchen Liu. Attention-driven training-free efficiency enhancement of diffusion models. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 16080–16089, 2024. 1, 2, 5, 6
- [45] Felix Wimbauer, Bichen Wu, Edgar Schoenfeld, Xiaoliang Dai, Ji Hou, Zijian He, Artsiom Sanakoyeu, Peizhao Zhang, Sam Tsai, Jonas Kohler, et al. Cache me if you can: Accelerating diffusion models through block caching. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 6211–6220, 2024. 1, 2
- [46] Xingyi Yang, Daquan Zhou, Jiashi Feng, and Xinchao Wang. Diffusion probabilistic model made slim. In Proceedings of the IEEE/CVF Conference on computer vision and pattern recognition, pages 22552–22562, 2023. 2
- [47] Jiahui Yu, Yuanzhong Xu, Jing Yu Koh, Thang Luong, Gunjan Baid, Zirui Wang, Vijay Vasudevan, Alexander Ku, Yinfei Yang, Burcu Karagol Ayan, et al. Scaling autoregressive models for content-rich text-to-image generation. arXiv preprint arXiv:2206.10789, 2(3):5, 2022. 6
- [48] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pages 3836–3847, 2023. 2
- [49] Tianchen Zhao, Tongcheng Fang, Enshu Liu, Wan Rui, Widayadewi Soedarmadji, Shiyao Li, Zinan Lin, Guohao Dai, Shengen Yan, Huazhong Yang, et al. Vidit-q: Efficient and accurate quantization of diffusion transformers for image and video generation. arXiv preprint arXiv:2406.02540, 2024. 2
- [50] Wangbo Zhao, Yizeng Han, Jiasheng Tang, Kai Wang, Yibing Song, Gao Huang, Fan Wang, and Yang You. Dynamic diffusion transformer. arXiv preprint arXiv:2410.03456, 2024. 1, 2
- [51] Zangwei Zheng, Xiangyu Peng, Tianji Yang, Chenhui Shen, Shenggui Li, Hongxin Liu, Yukun Zhou, Tianyi Li, and Yang You. Open-sora: Democratizing efficient video production for all. <https://github.com/hpcaitech/OpenSora>, 2024. 1, 2
- [52] Daquan Zhou, Weimin Wang, Hanshu Yan, Weiwei Lv, Yizhe Zhu, and Jiashi Feng. Magicvideo: Efficient video generation with latent diffusion models. arXiv preprint arXiv:2211.11018, 2022. 2