KEEP THE ESSENTIALS: EFFICIENT REFERENCE CONDITIONED GENERATION VIA TOKEN DROPPING

Anonymous authors
Paper under double-blind review

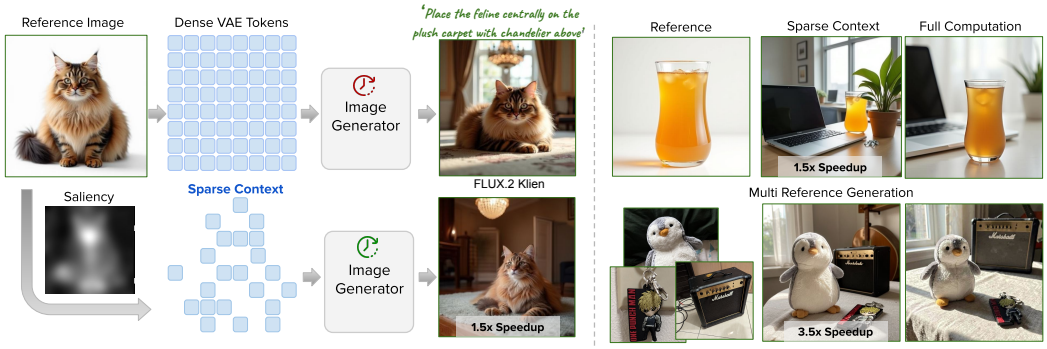


Figure 1: ***Sparse Context*** enhances the efficiency of reference-conditioned generation by replacing dense token grids with a compact set of sampled tokens, significantly accelerating inference in DiT-based reference-conditioned generative models.

ABSTRACT

Reference-based diffusion models enable highly controllable image generation by leveraging elements from input images to guide prompt-driven synthesis. However, these models are computationally expensive in runtime, and their cost scales severely with the number of input references. While the efficiency of diffusion models has been extensively studied in the context of prompt-driven generation, it remains largely under-explored in the realm of reference-based models. This setting presents unique challenges not addressed by methods focusing solely on generation. In particular, the wasteful representation of references as dense token grids offers significant opportunities for improvement. In this work, we present ***Sparse Context***, a method for constructing sparse reference representations by retaining only a reduced subset of reference tokens. We observe that even without modifying the model, dropping a significant portion of reference tokens at inference time largely preserves its generation capabilities. To fully realize this potential, we fine-tune the model with random token dropping at varying ratios, encouraging robustness to partial reference representations. Crucially, this training strategy decouples the model from any specific token selection rule, allowing flexible control at inference time. At inference time, instead of random dropping, we apply task-aware token selection strategies that prioritize the most informative regions of the reference images, adapting the token budget to the input and task requirements. Extensive experiments show our method achieves a $4\times$ increase in inference speed for multi-reference generation and an $2\times$ for single reference generation. Importantly, this efficiency is achieved without compromising visual quality across both spatially-aligned editing and subject-driven generation.

1 INTRODUCTION

Diffusion-based generative models have enabled remarkable progress in text-to-image/video synthesis, producing high-quality and diverse outputs from natural language descriptions (Rombach et al., 2022; Saharia et al., 2022; Dhariwal & Nichol, 2021; Wan et al., 2025). Beyond text-only generation, many practical applications require incorporating visual references, such as editing existing images (Meng et al., 2022; Hertz et al., 2022) or generating new scenes that preserve the identity

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

or style of a given subject (Ruiz et al., 2023; Gal et al., 2022). Reference-based diffusion models address this need by conditioning the generation process on one or more input images or videos, enabling direct visual control over the generated content (Batifol et al., 2025; MiniMax, 2026).

A common approach in recent transformer-based diffusion models (DiTs) Peebles & Xie (2023) is to represent reference images as tokens by encoding them with the model's VAE and concatenating the resulting latent tokens with those of the noisy input (Tan et al., 2024; Batifol et al., 2025). This representation is inherently dense, as each reference image is mapped to a full spatial grid of tokens. Since attention operations scale quadratically with the number of tokens, this substantially increases the cost of generation. As reference-based generation becomes increasingly prevalent in real-world applications, where multiple high-resolution references are often used to guide synthesis, this cost can quickly become a practical bottleneck. While efficiency has been extensively studied for prompt-driven diffusion models (Lu et al., 2025; Hu et al., 2025; Liu et al., 2025), reference-based models introduce additional computational challenges due to this dense representation, which also presents opportunities for improving efficiency.

We observe that full-resolution representation of reference images is often unnecessary. Reference images contain regions with varying levels of detail, where many tokens carry redundant information. Surprisingly, as illustrated in Fig. 2, even without modifying the model, dropping a large fraction of reference tokens during inference largely preserves generation quality with some loss in scene details. This suggests that reference-conditioned models are inherently robust to highly sparse representations.



Figure 2: **Redundancy in tokens of reference image.** We drop a large number of reference tokens for a pretrained reference-conditioned image generator. Even when dropping 80%, the output resembles the coarse layout of the input scene. This finding confirms reference tokens have high redundancy, which can be removed for efficiency.

Motivated by this observation, we introduce ***Sparse Context***, a framework for efficient reference-based diffusion that constructs sparse reference representations. We fine-tune the generative model to operate on partial reference inputs by randomly dropping tokens at varying ratios during training, encouraging robustness to different subsets of reference information. Importantly, this training strategy makes the model agnostic to the specific choice of tokens, enabling flexible inference-time control. At inference, we replace random dropping with task-aware token selection strategies that prioritize the most informative reference tokens, adapting the representation to the structure of each input and the requirements of different tasks. This results in substantial reductions in computational cost while maintaining high-quality generation.

We evaluate our approach on spatially aligned image editing and subject-driven generation tasks across several models. Importantly, our method also generalizes to video-to-video generation models via depth-to-video and edge-to-video tasks. By exploiting the inherent redundancy in reference-based representations, our method achieves substantial improvements in efficiency, accelerating inference by up to $2\times$ for a single reference and $4\times$ for multiple reference images, while preserving fine-grained visual quality. Importantly, our approach is orthogonal to existing efficiency techniques for diffusion models and can be readily combined with methods such as KV-caching and token merging(Bolya et al.) to further improve performance. Our results demonstrate that reference-based diffusion models can be made significantly more efficient without compromising their controllability or fidelity, highlighting the untapped efficiency potential inherent to this setting.

2 Related Works

Reference-conditioned generation extends diffusion models beyond text-only synthesis by incorporating visual inputs for improved control. However, adding reference images substantially increases computational cost due to the large number of conditioning tokens involved in attention. We review prior approaches for integrating such references into the diffusion process, along with methods for improving the efficiency of visual generative models.

2.1 Reference-based Image Generation

Standard text-to-image diffusion models (Rombach et al., 2022; Saharia et al., 2022) do not natively support image inputs. Consequently, dedicated methods have been developed to leverage existing images, either for image editing (Meng et al., 2022; Hertz et al., 2022; Parmar et al., 2023) or as references for subject- and style-driven generation (Gal et al., 2022; Ruiz et al., 2023). For image editing, a prominent line of work focuses on inference-time methods that operate on the latent noise corresponding to a given image (Cao et al., 2023; Tumanyan et al., 2022; Huang et al., 2025; Po et al., 2024; Parihar et al., 2025). When applied to real images, these approaches typically require an inversion process to map the image into the diffusion noise space (Mokady et al., 2023; Garibi et al., 2024; Huberman-Spiegelglas et al., 2024; Kulikov et al., 2025). To avoid this costly and often imperfect inversion process, alternative approaches generate synthetic edit pairs using such inference-time methods and subsequently train image-to-image models on the resulting data (Brooks et al., 2023; Sheynin et al., 2024; Parihar et al., 2026). In the context of subject- and style-driven generation, early methods relied on per-subject or per-style optimization procedures (Gal et al., 2022; Ruiz et al., 2023; Kumari et al., 2023). More advanced approaches instead learn encoders and modify the diffusion architecture to condition the denoising process on reference representations extracted from input images (Guo et al., 2024; xla, 2024; Ye et al., 2023; Wei et al., 2023; Parihar et al., 2024). A central challenge in training such models is the limited availability of large-scale datasets containing paired images of the same subject or style across diverse contexts.

Extended self-attention mechanisms have been shown to be effective for generating such paired data, benefiting both image editing and subject- or style-driven generation (Cao et al., 2023; Alaluf et al., 2023; Avrahami et al., 2025; Zeng et al., 2024; Gal et al., 2024; Shin et al., 2025). By enabling attention across multiple images, these methods promote consistency across generated outputs, for example when synthesizing the same object in different scenes. With the rise of transformer-based diffusion models (DiTs) (Peebles & Xie, 2023), a common strategy for incorporating image inputs is through conditioning tokens concatenated with the noisy latent tokens (Tan et al., 2024; Batifol et al., 2025; Kumari et al., 2025; Cai et al., 2025; Agrawal et al., 2026). This design allows the attention layers to enforce content preservation, whether maintaining subject identity or preserving input structure during editing. These conditioning tokens are often derived from a VAE encoder, facilitating alignment with the latent distribution used during denoising.

More recently, large-scale DiT-based foundation models have begun to natively support reference images as part of their input, integrating them through similar token concatenation mechanisms (Labs, 2025; Bai et al., 2025; Google DeepMind, 2025). This unifies text-to-image and image-to-image generation within a single model. Our work builds on such models to perform both image editing and subject-driven generation. We show that the attention operations, which constitute the dominant computational cost, can be made more efficient, as the relevant information can be compactly represented using a small number of conditioning tokens.

2.2 Efficient Visual Generation

Recent advancements in generative models have converged on transformer-based diffusion models (DiTs), which offer improved scalability and performance. However, this shift increases computational cost, as attention operations scale quadratically with the number of tokens. In particular, in reference-conditioned settings where input images are incorporated via token concatenation, the number of tokens can grow substantially. Conditioning on a single reference image effectively doubles the token count, with additional references further amplifying this cost.

A first line of work focuses on improving the efficiency of the attention operation itself, through sparse, linearized, or hardware-aware formulations (Beltagy et al., 2020; Choromanski et al., 2020; Katharopoulos et al., 2020; Shen et al., 2021; Han et al., 2023; Dao et al., 2022; Wang et al., 2020; Xia et al., 2025). A second direction reduces redundant computation by caching intermediate features across layers or timesteps, with key-value (KV) caching being a prominent example, enabling reuse of previously computed representations during the denoising process (Ma et al., 2024b; Wimbauer et al., 2024; Ma et al., 2024a; Liu et al., 2025; Kahatapitiya et al., 2025; Hu et al., 2025). A third line of work aims to reduce the number of tokens processed by the model, either by dropping or grouping tokens. Initially explored in Vision Transformers for image understanding (Bolya et al.; Kim et al., 2024), these approaches have more recently been adapted to generative settings with DiT architectures (Bolya & Hoffman, 2023; Lu et al., 2025; Wu et al., 2025; Hu et al., 2024).

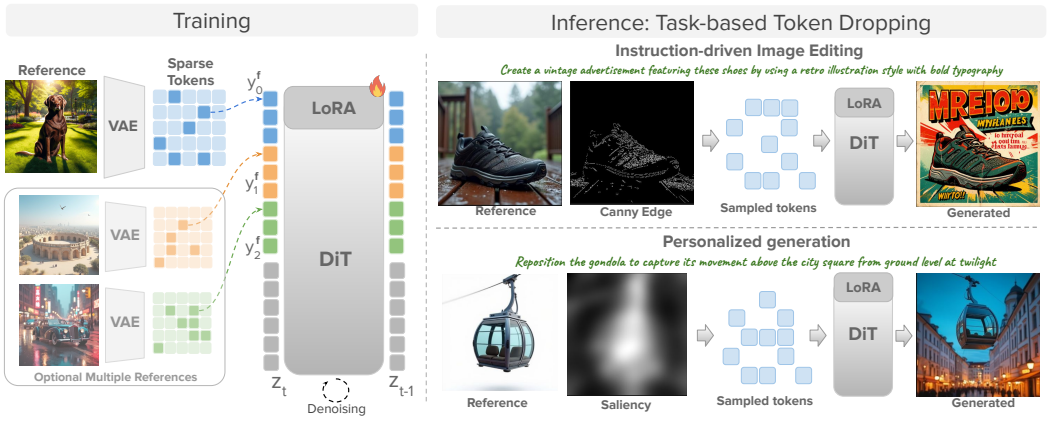


Figure 3: **Overview**. **Training:** During training, we randomly drop tokens from reference images at a keep fraction $f \in (0.05, 0.25)$, yielding a sparse token representation y_i^f that conditions the denoising of image tokens z_t. **Inference:** At inference time, users can drop reference tokens according to their compute budget and condition the DiT on the resulting sparse set for reference-based generation. Beyond random dropping, task-specific selection strategies can further improve quality under a fixed budget, such as sampling more tokens from edge regions to preserve structure during editing, or salient regions to better preserve identity during personalization.

While these methods are designed for general generative settings, reference-conditioned generation offers additional opportunities for efficiency. In this setting, the source image is fully observed, and its information can be compactly encoded rather than processed at full token resolution. In our work, we leverage this property to compress the reference into a small set of spatially-aware conditioning tokens, significantly reducing the cost of attention while preserving the relevant information.

3 Method

3.1 Preliminaries

Diffusion Models. Diffusion or flow-based models (Song et al., 2022; Lipman et al., 2023) learn to map a Gaussian noise distribution to an image distribution. In the forward diffusion process, an input image $x \in \mathcal{X}$ is corrupted with Gaussian noise $\epsilon \sim \mathcal{N}(0, I)$ to obtain a noisy version of the sample $x_t = x + \alpha_t \epsilon$, where α_t follows a predefined noise schedule. To learn the reverse mapping, a time-conditioned model $\epsilon_\theta(x_t, t)$ is trained to predict the added noise. For conditional generation, the denoising network is conditioned on an additional input signal y, i.e., $\epsilon_\theta(x_t, t, y)$, such as a text prompt for text-to-image generation or images for reference-conditioned generation.

Reference conditioned DiTs. State-of-the-art diffusion models Labs (2024) are implemented with Diffusion Image Transformer (DiT) (Peebles & Xie, 2023), where the images $x \in \mathcal{X}$ are encoded in VAE latent space and tokenized to obtain patch-level representations $z \in \mathcal{Z}$. These visual tokens are noised and processed by a sequence of DiT blocks, where self-attention enables interactions across all tokens. This token-based formulation naturally supports conditioning on image references, where the reference image is first tokenized y, and its tokens are concatenated to the noisy image tokens z_t. The self-attention mechanism enables strong conditioning between the reference images and the noisy latents, enabling high-quality reference-conditional generation.

3.2 Redundancy in Reference Image Tokens

A reference-conditioned generator takes as input a text prompt and one or multiple n reference images $\{y_i\}_{i=1}^n$, where $y_i \in \mathcal{X}$, to produce an edited image. The reference tokens are appended to the image latents x_t and processed by the DiT model. While powerful, this conditioning mechanism significantly increases computational cost, as self-attention scales quadratically with the number of input tokens. This issue becomes particularly severe when multiple references are used, as the number of reference tokens increases linearly with the number of reference images.

This raises a natural question: *do we actually need all reference image tokens?* To investigate this, we perform a simple experiment using the state-of-the-art generator FLUX.2-Klien (Labs, 2025) in which we randomly drop tokens from a reference image y_i with a keep fraction of $f \in \{0.10, 0.20\}$,

resulting in a sparse reference representation y_i^f at inference time. As shown in Fig. 2, even after removing 0.8 fraction of the reference tokens, the generated image still preserves the overall layout of the reference, although fine object and scene details are largely lost. This is a training-free setting, which we refer to as *Naïve Context Pruning*. This observation suggests that a significant portion of the reference tokens is redundant and can be removed. Bridging the remaining gap in fine details could therefore enable substantial acceleration at inference.

3.3 *Sparse Context*

We hypothesize that the performance gap between full-reference and partial-reference generation primarily arises from a train–test distribution mismatch induced by sparse reference conditioning at inference time. While the model is trained to condition on the full set of tokens y_i extracted from the reference image, it is used at inference time with only a subset of them y_i^f. To mitigate this mismatch, we fine-tune the generator to operate under sparse reference conditioning. During training, we randomly drop reference tokens with a keep fraction of $f \in (0.05, 0.25)$ before concatenating them with the noisy target tokens z_t shown in Fig. 3-left. The self-attention mechanism enables the model to effectively condition on the remaining reference tokens despite the reduced input.

In practice, we fine-tune a state-of-the-art reference-conditioned generator (Labs, 2025) using a lightweight LoRA on a curated dataset for reference-based image generation (Sec. 3.5). This adaptation substantially improves scene structure and identity preservation compared to the *Naïve Context Pruning* baseline (Fig. 6), supporting our hypothesis. Using a randomly sampled drop ratio during training encourages the model to operate robustly across different levels of sparsity in the reference tokens and enables flexible control over the inference-time compute budget while preserving essential reference information. At inference time, reference tokens can be randomly dropped according to the desired compute budget, allowing a direct trade-off between efficiency and generation quality.

3.4 Task-aware Token Dropping

Reference-conditioned models are used in a variety of settings, such as image editing, which requires spatially aligned generation, and personalization, where subjects from reference images are placed in new environments or combined into novel compositions. Our training strategy of randomly dropping tokens is agnostic to task-specific inductive biases, enabling flexible and general use during inference. In particular, it facilitates adapting to task-aware token dropping strategies at inference time across reference-conditioned generation tasks. We explore two representative tasks below and, for each, adopt an intuitive, task-specific token selection strategy rather than an exhaustive search. This is not a constraint of our framework, however: one can easily plug in a more sophisticated selection method in its place.

Instruction-driven Image Editing. Given a reference image and edit instructions in the form of a text prompt, the task is to generate an edited image that follows the specified instructions. A key requirement for consistent image editing is preserving the structure of the reference image while precisely modifying the desired aspects of the scene. In this regard, edge maps provide informative cues about scene structure, as they highlight object boundaries. To better preserve the input structure, we propose an edge-aware token sampling strategy, where more tokens are sampled from edge regions compared to uniform regions. Specifically, we first extract a Canny edge map from the reference image and use it as a probability distribution to sample reference tokens. As shown in Fig. 4, this approach better preserves scene structure than random dropping while maintaining accurate edits. Under the same token budget, edge-based sampling significantly improves structural consistency. As noted previously, this specific edge-based token rule is one instantiation of a pluggable component; we also explore a segmentation-mask-based token selection alternative for instruction-based image editing in the appendix, which retains at least k tokens from every image segment rather than only along edges (Appendix A.2), ensuring that small or low-contrast objects without strong edges are still preserved during editing.



Figure 4: **Token selection for image editing.** During inference, we use a Canny edge map to concentrate token selection on structural boundaries rather than random sampling. This edge-prioritized strategy more effectively preserves the underlying image structure and scene identity during the editing process.

Personalization. Given one or a few reference images, the task of personalization is to generate subjects from the reference images in new compositions. A key requirement for accurate personalization is preserving the identity of the subjects from the source images. In typical real-world images, the subject occupies only a small portion of the pixels. To this end, we use image saliency as an approximation of the subject mask in the reference image and employ it to perform weighted sampling of source tokens. As shown in Fig. 5, selecting tokens based on saliency significantly improves subject identity in the generated scene. Alternatively, segmentation maps of the input image can be used for more accurate subject localization.

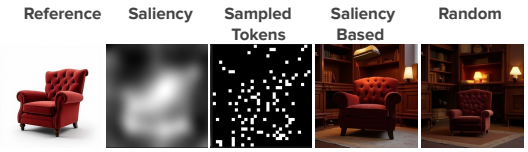


Figure 5: **Token Selection for Personalization.** We utilize a saliency map to localize the dominant scene object, allowing the selection process to concentrate on informative regions. By prioritizing these salient tokens, our method more effectively extracts fine-grained object details, leading to better identity preservation critical for high-fidelity image personalization.

3.5 Dataset

To train our model with token dropping, we construct a diverse dataset covering the main reference-conditioned tasks: instruction-driven editing and personalization. For instruction-driven editing, we use a $30K$ subset of HQ-Edit (Hui et al.), which contains high-quality source images and edit instructions. We regenerate the edited images using Flux-Klien2 (Labs, 2025), as the original edited images in HQ-Edit are not sufficiently high fidelity. For personalization, we use a $20K$ subset of the Subjects-200K dataset (Tan et al., 2024), which contains pairs of reference images of subjects on plain backgrounds and their corresponding compositions. In addition, for multi-reference generation, we generate $13K$ images, each containing compositions of 2-6 objects, using the CustomDiffusion-105 dataset (Kumari et al., 2023). We utilize a subset of $9,000$ videos from the OpenVid-1m (Nan et al., 2025) video dataset and synthetically generate the corresponding conditioning video streams (edge maps, depth maps, and grayscale frames) to generate a paired video-to-video dataset for training. Additional details are provided in the appendix.

4 Experiments

Implementation Details. We perform our experiments using the FLUX-2-Klein-9B (Labs, 2025) model and its step-distilled version with 8 inference steps, and a smaller FLUX-2-Klein-4B model. For the reference video-conditioned generation task, we use the MiniMax-H3 Base Ref2VA (MiniMax, 2026) model. For image models, all images used for training and evaluation are at a resolution of 512×512. Training is performed via efficient LoRA fine-tuning with rank 16, for a total of $15K$ iterations. We employ a curriculum training strategy, where the first $10K$ iterations use samples with a single reference image, and the remaining $5K$ iterations use a mixture of single-reference and multi-reference samples from our custom dataset, as described above. For the video model, we work with 384×384 resolution videos and perform fine-tuning with a LoRA rank of 16. We train with an effective batch size of 8 for image models and 1 for video model on 4 NVIDIA A100 GPUs.

Evaluation Dataset. For image editing, we use the standard and diverse PIE-Bench (Ju et al., 2023) benchmark, which consists of 700 source images, edit instructions, and edited images covering a wide range of editing operations, including stylization, object insertion/removal, and object replacement. For personalization, we evaluate on a subset of 500 examples from Subject-200K for single-reference personalization and on 200 compositional scenes from CustomDiffusion-105 using

Table 1: Evaluation of reference-based generation with different reference token fractions f.

	Instruction Driven Image Editing							Reference Based Personalization					
	Tok. f	**LPIPS ↓**	**CLIP-I ↑**	**DINO ↑**	**KID ↓** ($\times 10^{-3}$)	**CLIP T-I ↑**	**Speedup ↑**	**Tok. f**	**CLIP-I ↑**	**DINO ↑**	**CLIP T-I ↑**	**KID ↓** ($\times 10^{-3}$)	**Speedup ↑**
Baseline	1.00	0.433	0.833	0.760	1.5	0.315	1.00x	1.00	0.750	0.451	0.290	21.2	1.00x
Naïve	0.05	0.740	0.675	0.446	**4.5**	0.297	2.06x	0.05	0.654	0.318	**0.286**	28.6	2.06x
Ours	0.05	**0.665**	**0.772**	**0.660**	5.2	**0.304**	2.06x	0.05	**0.735**	**0.435**	0.275	**23.7**	2.06x
Naïve	0.10	0.719	0.723	0.553	3.5	0.309	1.83x	0.10	0.681	0.360	**0.287**	27.8	1.83x
Ours	0.10	**0.628**	**0.796**	**0.703**	**3.4**	**0.311**	1.83x	0.10	**0.748**	**0.452**	0.274	**22.7**	1.83x
Naïve	0.20	0.687	0.762	0.631	3.0	**0.316**	1.49x	0.20	0.692	0.390	**0.288**	27.1	1.49x
Ours	0.20	**0.576**	**0.817**	**0.738**	**2.4**	0.314	1.49x	0.20	**0.754**	**0.455**	0.274	**23.0**	1.49x



Figure 6: **Instruction-Driven Image Editing:** ***Sparse Context*** preserves the structure of the reference image during editing while accurately performing the described edit. **Personalization:** Our method preserves the object identity well even with only keeping 10% of reference tokens ($f = 0.1$). Naïve token dropping fails to preserve fine-grained object details such as the color of the rope.

multiple reference images (2–6 references per scene). For the video-to-video task, we evaluate on a custom test set of 100 videos from OpenVid-1M (Nan et al., 2025) dataset.

Evaluation Metrics. We report LPIPS (Zhang et al., 2018), CLIP-Image (CLIP-I) (Radford et al., 2021), and DINO (Oquab et al., 2023) similarity scores between the reference and generated images to measure structure and identity preservation. For personalization, we do not report LPIPS, as the generated outputs are not expected to be pixel-aligned with the reference images. To evaluate text-image alignment, we report the CLIP text-image similarity score (CLIP T-I). For PIE-Bench, we use the target image captions to compute the CLIP alignment score, while for personalization, we use the generation prompts. For the video-to-video task, we report video similarity using LPIPS and DINO-Image scores, and measure conditioning adherence by extracting Canny/depth/grayscale maps from the generated video and computing Edge/Depth/Gray-SSIM against the ground-truth conditioning metrics are averaged over frames. Finally, we report relative inference speedups averaged over 100 runs on a single NVIDIA $A100$ GPU, compared to the base model without token dropping.

4.1 Instruction-driven Image Editing

We present qualitative results in Fig. 6 and Appendix Fig. 12, comparing our method with *Naïve Context Pruning* (**Naïve**) and full-reference generation using all reference tokens (**Baseline**). Our method preserves the scene structure even when using only 10% of the reference tokens, while effectively performing the intended edits. In contrast, *Naïve Context Pruning* corrupts the scene identity, as seen in the teddy and fish examples in the first and third rows, respectively. On the efficiency front, we achieve a $1.49\times$ speedup using only 20% of the reference tokens, while maintaining quality comparable to generation with the full set of reference tokens. We also observe marginal improvements in scene structure when increasing the token budget from 10% to 20%. This provides users with inference-time control over the compute budget while achieving consistent image edits.

Importantly, reference-based generation is a one-to-many mapping, so pixel alignment with the baseline output ($f = 1.0$) is not a meaningful correctness measure: two valid outputs can differ substantially at the pixel level while both preserving identity and following the instruction. In the teddy bear example above, both the baseline and our method correctly perform the edit while preserving the teddy bear's identity, despite not being pixel-aligned; this divergence reflects valid generation diversity rather than a failure mode. This is also validated in our quantitative results in Tab. 1, where our method demonstrates significantly better identity preservation, reflected by lower LPIPS and higher DINO and CLIP-I scores compared to *Naïve Context Pruning*, while maintaining similar text-image alignment (CLIP T-I). Further, our method achieves more than a $2\times$ speedup at $f = 0.05$ keep fraction compared to the full-reference baseline with $f = 1.0$.

4.2 Personalization

Single reference. We present qualitative results for personalization in Fig. 6 and appendix Fig. 13. Our method achieves highly consistent personalization results, where the reference object is naturally integrated into the generated scene according to the text prompt, while preserving fine-grained subject details, despite using only a small fraction of the reference tokens. Our saliency-based token sampling successfully concentrates the representation on the object regions. In contrast, applying the same sampling strategy to the *Naïve Context Pruning* significantly distorts the subject identity, for example by incorrectly introducing human figures into paintings or altering vehicle structures. Quantitative results in Tab. 1 show that our method maintains strong identity preservation, reflected by high CLIP-I and DINO scores, while preserving text-image alignment. In practice, a CLIP T-I score above 0.26 generally indicates good prompt adherence, which is also evident in the qualitative results in Fig. 6-bottom.

Multiple references. Given multiple reference images (Fig. 6- bottom and appendix Fig. 14), our method generates natural scene compositions while accurately preserving object identities, such as the yellow flowers and background in the second row and the sofa in the last row. Since quantitatively measuring identity preservation in multi-reference generation is challenging, we evaluate the generated images using text-image alignment metrics (reported in the appendix) and present inference speedups across varying numbers of reference images in Fig. 7. Our speedups scale with the number of reference images, as reference tokens occupy most of the self-attention computation. For example, with $n = 6$ reference images, 86% of the tokens correspond to reference images, while only 14% correspond to the target image. ***Sparse Context*** removes redundant reference tokens, yielding substantial speedups. In particular, for more than five reference images, we achieve over $4\times$ speedups with keep fractions of $f = 0.10$ & 0.05.

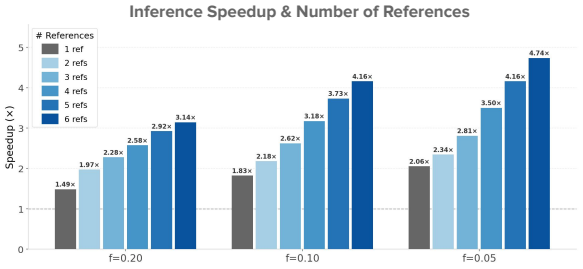


Figure 7: Speedup for multi-reference personalization across different drop fractions.

4.3 Baseline Comparison

We compare our method against existing inference-time token reduction approaches, originally designed for image understanding with ViTs and for DiT-based text-to-image generation. These approaches merge similar tokens to reduce the total token count. We consider the following baselines: a) **Token Averaging (Token Avg.)**: a naive approach that reduces token count by averaging tokens

Table 2: Comparison with SoTA token reduction methods (at $f = 0.20$).

	Instruction Driven Image Editing						Reference Based Personalization				
	Tok. f	LPIPS ↓	CLIP-I ↑	DINO ↑	KID ↓ ($\times 10^{-3}$)	CLIP T-I ↑	Tok. f	CLIP-I ↑	DINO ↑	CLIP T-I ↑	KID ↓ ($\times 10^{-3}$)
Baseline	1.00	0.433	0.833	0.760	1.5	0.315	1.00	0.750	0.451	0.290	21.2
Naïve	0.20	0.687	0.762	0.631	3.0	**0.316**	0.20	0.692	0.390	**0.288**	27.0
Token Avg	0.20	0.662	0.735	0.598	2.4	0.313	0.20	0.673	0.355	**0.288**	28.7
ToMe Bolya et al.	0.20	0.637	0.767	0.653	1.5	0.315	0.20	0.687	0.377	**0.288**	27.5
ToMA Lu et al. (2025)	0.20	0.646	0.771	0.665	**1.4**	**0.316**	0.20	0.696	0.393	**0.288**	28.6
Ours	0.20	**0.576**	**0.817**	**0.738**	2.4	0.314	0.20	**0.754**	**0.455**	0.274	**23.0**

within every $k \times k$ patch in the latent space; we choose k to match the reference token fraction. b) **Token Merging (ToMe)** (Bolya et al.): originally designed for ViTs, merging tokens based on patch similarity; in our setting, we apply this only once, after encoding the reference images through the VAE encoder, consistent with our method's one-shot design. c) **Token Merging with Attention (ToMA)** (Lu et al., 2025): designed for text-to-image generation, this method merges intermediate image tokens via self-attention-based patch similarity; we integrate their algorithm with the reference image tokens. Our method achieves substantially better image and identity preservation than all three baselines (lower LPIPS, higher DINO) while accurately performing the desired edit; qualitative results are in Appendix Fig. 10. Notably, all three baselines can be readily combined with ***Sparse Context*** at inference time for further improvements (results in Appendix Tab. 4).

4.4 Reference Conditioned Video Generation

We demonstrate that ***Sparse Context*** extends directly to video by implementing it on MiniMax-H3 (MiniMax, 2026), a recent DiT-based reference-conditioned video generation model. MiniMax-H3 maps an input conditioning video and a text prompt to an output video that follows both, using the same mechanism as reference-conditioned image models: the reference video is tokenized into spatio-temporal tokens, which are concatenated with the noisy video tokens to condition generation. Because ***Sparse Context*** operates on this set of reference tokens, it applies without modification to this architecture. We evaluate this on three representative video-to-video tasks: edge-to-video, depth-to-video, and grayscale-to-video, where the reference is a spatially dense conditioning video (edge, depth, or grayscale) and the text prompt describes only the target content, since spatial structure is already fully specified by the conditioning video. We present results in Tab. 3 and Fig. 11. As shown in Table 3, our method achieves substantial speedups while producing videos close to the ground truth that maintain strict structural alignment with the conditioning inputs (higher SSIM). These results confirm that our method generalizes beyond the image domain to video generation. Check the attached .html page for visualizing the video results.

Table 3: Efficiency on reference-conditioned video generation with the MiniMax-H3 model.

		Canny → Video			Gray → Video			Depth → Video			
Setting	Tok. f	LPIPS ↓	DINO ↑	Edge-SSIM ↑	LPIPS ↓	DINO ↑	Gray-SSIM ↑	LPIPS ↓	DINO ↑	Depth-SSIM ↑	Speedup ↑
Baseline	1.00	0.461	0.752	0.535	0.411	0.823	0.592	0.531	0.710	0.833	1.00×
Naïve	0.10	0.549	0.695	0.467	0.550	0.748	0.434	0.607	0.649	0.750	1.42×
Ours	0.10	**0.514**	**0.709**	**0.513**	**0.467**	**0.781**	**0.524**	**0.580**	**0.654**	**0.789**	1.42×
Naïve	0.20	0.516	0.709	0.491	0.500	0.772	0.478	0.578	**0.665**	0.788	1.50×
Ours	0.20	**0.486**	**0.716**	**0.530**	**0.399**	**0.809**	**0.608**	**0.564**	0.660	**0.806**	1.50×

5 Conclusion and Discussion

We propose ***Sparse Context***, a simple token-dropping strategy for efficient reference-conditioned image and video generation. We show that most tokens originating from reference are redundant and can be discarded, significantly reducing the quadratic cost of the self-attention mechanism. However, naïvely dropping tokens at inference time distorts scene identity due to the mismatch between the training and inference distributions. To address this, we perform lightweight post-training on a curated dataset consisting of two reference-conditioned generation tasks. The resulting model is substantially more efficient at inference time, achieving up to $5\times$ speedups for multi-reference generation. Furthermore, our generalized retraining strategy enables flexible inference-time token dropping schemes tailored to specific tasks, such as saliency-guided token selection for reference-based generation. Our method is complementary to existing efficiency techniques, such as KV-caching and token merging, and can be combined with them for additional gains. More broadly, our findings highlight a significant source of redundancy in contemporary DiT-based image and video generation models, which are otherwise extremely expensive at high resolutions and with multiple references.

AI USE STATEMENT

In this work, we used AI tools to assist with parts of our method and implementation and baseline comparison, to format tables, and to improve the writing of the paper. We also used AI tools to help gather related work and approaches from the literature for comparison with our method. This assistance was limited to coding tools and LLMs used for paraphrasing and editing text at the paragraph level. We did not use generative AI to generate ideas, iterate on our method, or interpret results and key findings. We have reviewed all AI-assisted work and take full responsibility for it.

REFERENCES

Xlabs-ai flux ip-adapter. https://huggingface.co/XLabs-AI/flux-ip-adapter, 2024.

Vaibhav Agrawal, Rishubh Parihar, Pradhaan S Bhat, Ravi Kiran Sarvadevabhatla, and Venkatesh Babu Radhakrishnan. Seethrough3d: Occlusion aware 3d control in text-to-image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 25403–25414, 2026.

Yuval Alaluf, Daniel Garibi, Or Patashnik, Hadar Averbuch-Elor, and Daniel Cohen-Or. Cross-image attention for zero-shot appearance transfer, 2023. URL https://arxiv.org/abs/2311.03335.

Omri Avrahami, Or Patashnik, Ohad Fried, Egor Nemchinov, Kfir Aberman, Dani Lischinski, and Daniel Cohen-Or. Stable flow: Vital layers for training-free image editing. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 7877–7888, 2025.

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, Jiabo Ye, Xi Zhang, Tianbao Xie, Zesen Cheng, Hang Zhang, Zhibo Yang, Haiyang Xu, and Junyang Lin. Qwen2.5-vl technical report. *arXiv preprint arXiv:2502.13923*, 2025.

Stephen Batifol, Andreas Blattmann, Frederic Boesel, Saksham Consul, Cyril Diagne, Tim Dockhorn, Jack English, Zion English, Patrick Esser, Sumith Kulal, et al. Flux. 1 kontext: Flow matching for in-context image generation and editing in latent space. *arXiv e-prints*, pp. arXiv–2506, 2025.

Iz Beltagy, Matthew E Peters, and Arman Cohan. Longformer: The long-document transformer. *arXiv preprint arXiv:2004.05150*, 2020.

Daniel Bolya and Judy Hoffman. Token merging for fast stable diffusion. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 4599–4603, 2023.

Daniel Bolya, Cheng-Yang Fu, Xiaoliang Dai, Peizhao Zhang, Christoph Feichtenhofer, and Judy Hoffman. Token merging: Your vit but faster. In *The Eleventh International Conference on Learning Representations*.

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

Shengqu Cai, Eric Ryan Chan, Yunzhi Zhang, Leonidas Guibas, Jiajun Wu, and Gordon Wetzstein. Diffusion self-distillation for zero-shot customized image generation. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 18434–18443, 2025.

Mingdeng Cao, Xintao Wang, Zhongang Qi, Ying Shan, Xiaohu Qie, and Yinqiang Zheng. Masactrl: Tuning-free mutual self-attention control for consistent image synthesis and editing, 2023. URL https://arxiv.org/abs/2304.08465.

Krzysztof Choromanski, Valerii Likhosherstov, David Dohan, Xingyou Song, Andreea Gane, Tamas Sarlos, Peter Hawkins, Jared Davis, Afroz Mohiuddin, Lukasz Kaiser, et al. Rethinking attention with performers. *arXiv preprint arXiv:2009.14794*, 2020.

Tri Dao, Dan Fu, Stefano Ermon, Atri Rudra, and Christopher Ré. Flashattention: Fast and memory-efficient exact attention with io-awareness. *Advances in neural information processing systems*, 35:16344–16359, 2022.

Prafulla Dhariwal and Alex Nichol. Diffusion models beat gans on image synthesis, 2021. URL https://arxiv.org/abs/2105.05233.

Rinon Gal, Yuval Alaluf, Yuval Atzmon, Or Patashnik, Amit H. Bermano, Gal Chechik, and Daniel Cohen-Or. An image is worth one word: Personalizing text-to-image generation using textual inversion, 2022. URL https://arxiv.org/abs/2208.01618.

Rinon Gal, Or Lichter, Elad Richardson, Or Patashnik, Amit H. Bermano, Gal Chechik, and Daniel Cohen-Or. Lcm-lookahead for encoder-based text-to-image personalization, 2024.

Daniel Garibi, Or Patashnik, Andrey Voynov, Hadar Averbuch-Elor, and Daniel Cohen-Or. Renoise: Real image inversion through iterative noising. *arXiv preprint arXiv:2403.14602*, 2024.

Google DeepMind. Gemini image - nano banana, 2025. URL https://deepmind.google/models/gemini-image/.

Zinan Guo, Yanze Wu, Zhuowei Chen, Lang Chen, Peng Zhang, and Qian He. Pulid: Pure and lightning id customization via contrastive alignment. In *Advances in Neural Information Processing Systems*, 2024.

Dongchen Han, Xuran Pan, Yizeng Han, Shiji Song, and Gao Huang. Flatten transformer: Vision transformer using focused linear attention. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 5961–5971, 2023.

Amir Hertz, Ron Mokady, Jay Tenenbaum, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Prompt-to-prompt image editing with cross attention control. 2022.

Taihang Hu, Linxuan Li, Joost Van de Weijer, Hongcheng Gao, Fahad S Khan, Jian Yang, Ming-Ming Cheng, Kai Wang, and Yaxing Wang. Token merging for training-free semantic binding in text-to-image synthesis. *Advances in Neural Information Processing Systems*, 37:137646–137672, 2024.

Zhanqiu Hu, Jian Meng, Yash Akhauri, Mohamed S Abdelfattah, Jae-sun Seo, Zhiru Zhang, and Udit Gupta. Flashdlm: Accelerating diffusion language model inference via efficient kv caching and guided diffusion. *arXiv preprint arXiv:2505.21467*, 2025.

Yi Huang, Jiancheng Huang, Yifan Liu, Mingfu Yan, Jiaxi Lv, Jianzhuang Liu, Wei Xiong, He Zhang, Liangliang Cao, and Shifeng Chen. Diffusion model-based image editing: A survey. *IEEE Trans. Pattern Anal. Mach. Intell.*, 47(6):4409–4437, June 2025. ISSN 0162-8828. doi: 10.1109/TPAMI.2025.3541625. URL https://doi.org/10.1109/TPAMI.2025.3541625.

Inbar Huberman-Spiegelglas, Vladimir Kulikov, and Tomer Michaeli. An edit friendly ddpm noise space: Inversion and manipulations, 2024. URL https://arxiv.org/abs/2304.06140.

Mude Hui, Siwei Yang, Bingchen Zhao, Yichun Shi, Heng Wang, Peng Wang, Cihang Xie, and Yuyin Zhou. Hq-edit: A high-quality dataset for instruction-based image editing. In *The Thirteenth International Conference on Learning Representations*.

Xuan Ju, Ailing Zeng, Yuxuan Bian, Shaoteng Liu, and Qiang Xu. Pnp inversion: Boosting diffusion-based editing with 3 lines of code. In *The Twelfth International Conference on Learning Representations*, 2023.

Kumara Kahatapitiya, Haozhe Liu, Sen He, Ding Liu, Menglin Jia, Chenyang Zhang, Michael S Ryoo, and Tian Xie. Adaptive caching for faster video generation with diffusion transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 15240–15252, 2025.

Angelos Katharopoulos, Apoorv Vyas, Nikolaos Pappas, and François Fleuret. Transformers are rnns: Fast autoregressive transformers with linear attention. In *International conference on machine learning*, pp. 5156–5165. PMLR, 2020.

Minchul Kim, Shangqian Gao, Yen-Chang Hsu, Yilin Shen, and Hongxia Jin. Token fusion: Bridging the gap between token pruning and token merging. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 1383–1392, 2024.

Vladimir Kulikov, Matan Kleiner, Inbar Huberman-Spiegelglas, and Tomer Michaeli. Flowedit: Inversion-free text-based editing using pre-trained flow models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 19721–19730, 2025.

Nupur Kumari, Bingliang Zhang, Richard Zhang, Eli Shechtman, and Jun-Yan Zhu. Multi-concept customization of text-to-image diffusion. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 1931–1941, 2023.

Nupur Kumari, Xi Yin, Jun-Yan Zhu, Ishan Misra, and Samaneh Azadi. Generating multi-image synthetic data for text-to-image customization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 16524–16534, 2025.

Matthias Kümmerer, Harneet Singh Khanuja, and Matthias Bethge. Modeling saliency dataset bias. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 22077–22088, 2025.

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

Black Forest Labs. FLUX.2: Frontier Visual Intelligence. `https://bfl.ai/blog/flux-2`, 2025.

Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matthew Le. Flow matching for generative modeling. In *The Eleventh International Conference on Learning Representations*, 2023.

Feng Liu, Shiwei Zhang, Xiaofeng Wang, Yujie Wei, Haonan Qiu, Yuzhong Zhao, Yingya Zhang, Qixiang Ye, and Fang Wan. Timestep embedding tells: It's time to cache for video diffusion model. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 7353–7363, 2025.

Wenbo Lu, Shaoyi Zheng, Yuxuan Xia, and Shengjie Wang. Toma: Token merge with attention for diffusion models. In *International Conference on Machine Learning*, pp. 40930–40951. PMLR, 2025.

Xinyin Ma, Gongfan Fang, Michael Bi Mi, and Xinchao Wang. Learning-to-cache: Accelerating diffusion transformer via layer caching. *Advances in Neural Information Processing Systems*, 37: 133282–133304, 2024a.

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*, pp. 15762–15772, 2024b.

Chenlin Meng, Yutong He, Yang Song, Jiaming Song, Jiajun Wu, Jun-Yan Zhu, and Stefano Ermon. Sdedit: Guided image synthesis and editing with stochastic differential equations, 2022. URL `https://arxiv.org/abs/2108.01073`.

MiniMax. MiniMax-H3. `https://github.com/MiniMax-AI/MiniMax-H3`, 2026.

Ron Mokady, Amir Hertz, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Null-text inversion for editing real images using guided diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6038–6047, June 2023.

Kepan Nan, Rui Xie, Penghao Zhou, Tiehan Fan, Zhenheng Yang, Zhijie Chen, Xiang Li, Jian Yang, and Ying Tai. Openvid-1m: A large-scale high-quality dataset for text-to-video generation. In *International conference on learning representations*, volume 2025, pp. 1045–1064, 2025.

Maxime Oquab, Timothée Darcet, Theo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Russell Howes, Po-Yao Huang, Hu Xu, Vasu Sharma, Shang-Wen Li, Wojciech Galuba, Mike Rabbat, Mido Assran, Nicolas Ballas, Gabriel Synnaeve, Ishan Misra, Herve Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. Dinov2: Learning robust visual features without supervision, 2023.

Rishubh Parihar, VS Sachidanand, Sabariswaran Mani, Tejan Karmali, and R Venkatesh Babu. Precisecontrol: Enhancing text-to-image diffusion models with fine-grained attribute control. In *European Conference on Computer Vision*, pp. 469–487. Springer, 2024.

Rishubh Parihar, Sachidanand VS, and R Venkatesh Babu. Zero-shot depth aware image editing with diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 15748–15759, 2025.

Rishubh Parihar, Or Patashnik, Daniil Ostashev, Venkatesh Babu Radhakrishnan, Daniel Cohen-Or, and Kuan-Chieh Jackson Wang. Kontinuous kontext: Continuous strength control for instruction-based image editing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 37929–37939, 2026.

Gaurav Parmar, Krishna Kumar Singh, Richard Zhang, Yijun Li, Jingwan Lu, and Jun-Yan Zhu. Zero-shot image-to-image translation. In *Special Interest Group on Computer Graphics and Interactive Techniques Conference Conference Proceedings*, SIGGRAPH '23, pp. 1–11. ACM, July 2023. doi: 10.1145/3588432.3591513. URL http://dx.doi.org/10.1145/3588432.3591513.

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

R. Po, W. Yifan, V. Golyanik, K. Aberman, J. T. Barron, A. Bermano, E. Chan, T. Dekel, A. Holynski, A. Kanazawa, C.K. Liu, L. Liu, B. Mildenhall, M. Nießner, B. Ommer, C. Theobalt, P. Wonka, and G. Wetzstein. State of the art on diffusion models for visual computing. *Computer Graphics Forum*, 43(2):e15063, 2024. doi: https://doi.org/10.1111/cgf.15063. URL https://onlinelibrary.wiley.com/doi/abs/10.1111/cgf.15063.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners.

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.

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 (CVPR)*, pp. 10684–10695, June 2022.

Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Yael Pritch, Michael Rubinstein, and Kfir Aberman. Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 22500–22510, 2023.

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

Zhuoran Shen, Mingyuan Zhang, Haiyu Zhao, Shuai Yi, and Hongsheng Li. Efficient attention: Attention with linear complexities. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pp. 3531–3539, 2021.

Shelly Sheynin, Adam Polyak, Uriel Singer, Yuval Kirstain, Amit Zohar, Oron Ashual, Devi Parikh, and Yaniv Taigman. Emu edit: Precise image editing via recognition and generation tasks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8871–8879, 2024.

Chaehun Shin, Jooyoung Choi, Heeseung Kim, and Sungroh Yoon. Large-scale text-to-image model with inpainting is a zero-shot subject-driven image generator. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 7986–7996, 2025.

Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models, 2022. URL https://arxiv.org/abs/2010.02502.

Zhenxiong Tan, Songhua Liu, Xingyi Yang, Qiaochu Xue, and Xinchao Wang. Ominicontrol: Minimal and universal control for diffusion transformer. *arXiv preprint arXiv:2411.15098*, 2024.

Narek Tumanyan, Michal Geyer, Shai Bagon, and Tali Dekel. Plug-and-play diffusion features for text-driven image-to-image translation, 2022. URL https://arxiv.org/abs/2211.12572.

Team Wan, Ang Wang, Baole Ai, Bin Wen, Chaojie Mao, Chen-Wei Xie, Di Chen, Feiwu Yu, Haiming Zhao, Jianxiao Yang, Jianyuan Zeng, Jiayu Wang, Jingfeng Zhang, Jingren Zhou, Jinkai Wang, Jixuan Chen, Kai Zhu, Kang Zhao, Keyu Yan, Lianghua Huang, Mengyang Feng, Ningyi Zhang, Pandeng Li, Pingyu Wu, Ruihang Chu, Ruili Feng, Shiwei Zhang, Siyang Sun, Tao Fang, Tianxing Wang, Tianyi Gui, Tingyu Weng, Tong Shen, Wei Lin, Wei Wang, Wei Wang, Wenmeng Zhou, Wente Wang, Wenting Shen, Wenyuan Yu, Xianzhong Shi, Xiaoming Huang, Xin Xu, Yan Kou, Yangyu Lv, Yifei Li, Yijing Liu, Yiming Wang, Yingya Zhang, Yitong Huang, Yong Li, You Wu, Yu Liu, Yulin Pan, Yun Zheng, Yuntao Hong, Yupeng Shi, Yutong Feng, Zeyinzi Jiang, Zhen Han, Zhi-Fan Wu, and Ziyu Liu. Wan: Open and advanced large-scale video generative models. *arXiv preprint arXiv:2503.20314*, 2025.

Sinong Wang, Belinda Z Li, Madian Khabsa, Han Fang, and Hao Ma. Linformer: Self-attention with linear complexity. *arXiv preprint arXiv:2006.04768*, 2020.

Yuxiang Wei, Yabo Zhang, Zhilong Ji, Jinfeng Bai, Lei Zhang, and Wangmeng Zuo. Elite: Encoding visual concepts into textual embeddings for customized text-to-image generation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 15943–15953, 2023.

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*, pp. 6211–6220, 2024.

Haoyu Wu, Jingyi Xu, Hieu Le, and Dimitris Samaras. Importance-based token merging for efficient image and video generation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4983–4995, 2025.

Yifei Xia, Suhan Ling, Fangcheng Fu, Yujie Wang, Huixia Li, Xuefeng Xiao, and Bin Cui. Training-free and adaptive sparse attention for efficient long video generation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 15982–15993, 2025.

Lihe Yang, Bingyi Kang, Zilong Huang, Zhen Zhao, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth anything v2. *arXiv:2406.09414*, 2024.

Hu Ye, Jun Zhang, Sibo Liu, Xiao Han, and Wei Yang. Ip-adapter: Text compatible image prompt adapter for text-to-image diffusion models, 2023. URL https://arxiv.org/abs/2308.06721.

Yu Zeng, Vishal M Patel, Haochen Wang, Xun Huang, Ting-Chun Wang, Ming-Yu Liu, and Yogesh Balaji. Jedi: Joint-image diffusion models for finetuning-free personalized text-to-image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024.

Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 586–595, 2018.

A SUPPLEMENTARY MATERIAL

A.1 INTEGRATING *Sparse Context* WITH OTHER EFFICIENCY METHODS

Our proposed method for efficient generation is simple and can easily be plugged into existing efficiency approaches. Here we show the integration of our method with KV Caching and integration with other Token Merging methods.

KV-cache: KV-cache Radford et al. is one of the widely used techniques to improve the efficiency of attention-based models. The idea is to store the key and values from the first computation and later retrieve the stored key and values for the attention computation. In the context of reference-based image generation with diffusion models, the KV of the reference tokens is stored during the first denoising step and then retrieved and used for subsequent steps. This provides substantial inference time speedups, and similar to our approach, the gains improve with the number of reference images. Our method and KV-cache explore complementary directions for efficiency, and KV-cache can be integrated with our method to achieve compounded efficiency gains. To this end, we implement the KV-cache on our selected reference tokens for the multi-reference generation task with $f = 0.10$, and the results are presented in Fig. 8. Notably, our method, when integrated with KV-cache, achieves substantial speedup up to $6.72\times$ when prompted with 6 reference images.

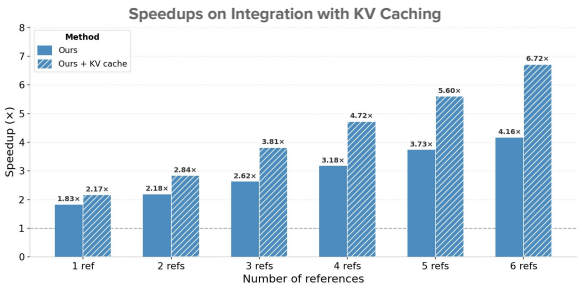


Figure 8: Our method is complementary to KV-caching and can be easily integrated to achieve further speedup gains for reference-conditioned generation.

Token merging: We integrate our method with existing inference token merging approaches in Tab. 4. Specifically, we use our trained LoRA for generation along with different token reduction methods. Instead of selecting tokens to drop based on saliency or Canny maps, we can naively perform token averaging with our trained checkpoint. Alternatively, one can use token similarity to merge tokens, rather than task-aware selection, at inference time. The results show that our method improves performance over the Naïve baseline across all three token reduction strategies: our task-aware token selection, token averaging, and token merging. This demonstrates that our pretraining generalizes well and can be readily adapted to different token reduction methods at inference time.

Table 4: ***Sparse Context***seamlessly integrates with the existing token merging method by merging the reference tokens to reduce token count instead of token dropping during inference. Our generalized pretraining adapts the model to infer reference image information with small number of tokens agnostic to the type of token reduction method used.

	Reference Based Personalization					Instruction Driven Image Editing					
	Tok. f	**CLIP-I** ↑	**DINO** ↑	**CLIP T-I** ↑	**KID** ↓ ($\times 10^{-3}$)	**Tok.** f	**LPIPS** ↓	**CLIP-I** ↑	**DINO** ↑	**CLIP T-I** ↑	**KID** ↓ ($\times 10^{-3}$)
Baseline	1.00	0.750	0.451	0.290	21.23	1.00	0.433	0.833	0.760	0.315	1.50
Naïve	0.20	0.692	0.390	**0.288**	27.10	0.20	0.687	0.762	0.631	**0.316**	2.97
Ours	0.20	**0.754**	**0.455**	0.274	**23.01**	0.20	0.576	0.817	0.738	0.314	2.37
Token Avg + Ours	0.20	0.735	0.427	0.274	25.19	0.20	0.605	0.791	0.712	0.314	1.48
ToMe + Ours	0.20	0.747	0.448	0.274	25.10	0.20	**0.529**	**0.818**	**0.767**	**0.316**	**1.36**

A.2 ALTERNATIVES TASK-AWARE TOKEN SELECTION

We emphasize that our generalized training with random token dropping enables inference-time flexibility to select any token-selection-based method suited to the use case. In our experiments, we chose Canny edges for the image editing task and saliency maps for the personalization task, as intuitive choices for token selection rather than performing an exhaustive search. However, this is not a constraint of our framework: one can easily plug in a better task-specific token selection method, as we demonstrate below.

Segmentation-based token selection: To demonstrate this, we additionally evaluate segmentation-mask-based selection, which guarantees that at least some tokens are retained from every segment rather than only along edges (Tab. 5). This outperforms Canny-based selection, confirming that our framework benefits directly from more advanced selection strategies. We did not optimize the selection heuristic itself, and intentionally leave it as a pluggable component so users can substitute stronger task-specific models (e.g., segmentation, saliency, or future learned selectors) as needed.

Table 5: Effect of the token selection strategy on image editing quality at a fixed reference token fraction $f = 0.20$. Segmentation-based sampling, which guarantees coverage of every object segment, outperforms Canny-edge-based selection, showing that our framework benefits directly from stronger task-specific selection rules.

Setting	Tok. f	LPIPS ↓	CLIP-I ↑	DINO ↑	CLIP T-I ↑	KID ↓ ($\times 10^{-3}$)
Baseline	1.00	0.433	0.833	0.760	0.315	1.5
Naïve (Canny)	0.20	0.687	0.762	0.631	**0.316**	2.97
Ours (Canny)	0.20	0.576	0.817	0.738	0.314	2.37
Segmentation Sampling	0.20	**0.510**	**0.824**	**0.778**	**0.316**	**1.26**

A.3 Generalization to higher resolution images

We evaluate our method at 1024×1024 resolution with a reference-token keep fraction of 0.20 in Tab. 6. Our method substantially outperforms the naive baseline on identity preservation: LPIPS 0.627 (ours) vs. 0.706 (naive) for image editing, and CLIP-I 0.773 (ours) vs. 0.716 (naive) for personalization and is comparable to the No-Drop (full-reference) baseline on personalization. Notably, the speedup also improves with resolution: 1.79× at 1024 resolution vs. 1.49× at 512 resolution for the same 0.20 keep fraction, confirming that the efficiency gains from our method increase as resolution scales due to quadratic increase in reference tokens with resolution.

Table 6: Generalization of ***Sparse Context*** on high resolution 1024x1024 images

	Reference Based Personalization						Instruction Driven Image Editing						
	Tok. f	CLIP-I ↑	DINO ↑	CLIP T-I ↑	KID ↓ ($\times 10^{-3}$)	Speedup ↑	Tok. f	LPIPS ↓	CLIP-I ↑	DINO ↑	CLIP T-I ↑	KID ↓ ($\times 10^{-3}$)	Speedup ↑
Baseline	1.00	0.733	0.467	0.280	20.12	1.00x	1.00	0.534	0.777	0.668	0.327	5.25	1.00x
Naïve	0.20	0.716	0.433	**0.280**	**22.79**	1.79x	0.20	0.706	0.728	0.543	**0.325**	6.18	1.79x
Ours	0.20	**0.773**	**0.482**	0.271	25.50	1.79x	0.20	**0.627**	**0.778**	**0.640**	0.324	**4.84**	1.79x

A.4 ***Sparse Context*** for Additional Models

We additionally present results of our method on three different Flux models: i) **Flux2-Klien-9B-base** - a reference conditioned image generation model trained for reference image conditioned generation, ii) **Step distilled Flux2-Klien-9B** model inferred with 8 inference steps, and iii) a smaller **Flux2-Klien-4B-base**. The results are shown in Tab. 7, 8 and Tab.9. Our fine-tuning strategy significantly outperforms the Naïve token dropping strategy in terms of image fidelity and text alignment while being equally fast. For the base model, our method achieves more than $2\times$ speedup when 5% of the tokens are preserved. This showcases the generalization of our approach in accelerating different model architectures, including faster few-step inference methods.

Table 7: **Flux2-Klien-9B-base** Evaluation of generation quality with SparseContext.

	Instruction Driven Image Editing							Reference Based Personalization					
	Tok. f	LPIPS ↓	CLIP-I ↑	DINO ↑	KID ↓ ($\times 10^{-3}$)	CLIP T-I ↑	Speedup ↑	Tok. f	CLIP-I ↑	DINO ↑	CLIP T-I ↑	KID ↓ ($\times 10^{-3}$)	Speedup ↑
Baseline	1.00	0.414	0.827	0.758	1.3	0.321	1.00x	1.00	0.727	0.416	0.284	27.1	1.00x
Naïve	0.05	0.759	0.657	0.388	9.3	0.293	2.21x	0.05	0.674	0.328	0.2830	**24.7**	2.21x
Ours	0.05	**0.688**	**0.747**	**0.592**	**3.1**	**0.314**	2.21x	0.05	**0.718**	**0.396**	**0.2833**	27.2	2.21x
Naïve	0.10	0.740	0.707	0.497	4.5	0.306	1.88x	0.10	0.700	0.372	0.2838	**24.7**	1.88x
Ours	0.10	**0.646**	**0.773**	**0.645**	**2.6**	**0.319**	1.88x	0.10	**0.728**	**0.416**	**0.2839**	27.5	1.88x
Naïve	0.20	0.707	0.753	0.595	2.9	0.315	1.66x	0.20	0.717	0.400	**0.285**	**25.5**	1.66x
Ours	0.20	**0.596**	**0.793**	**0.686**	**2.3**	**0.321**	1.66x	0.20	**0.735**	**0.424**	0.284	29.2	1.66x

Table 8: **Step distilled (8-step)** inference on Flux2-Klien-9B evaluation of generation quality

	Instruction Driven Image Editing							Reference Based Personalization					
	Tok. f	LPIPS ↓	CLIP-I ↑	DINO ↑	KID ↓ ($\times 10^{-3}$)	CLIP T-I ↑	Speedup ↑	Tok. f	CLIP-I ↑	DINO ↑	CLIP T-I ↑	KID ↓ ($\times 10^{-3}$)	Speedup ↑
Baseline	1.00	0.409	0.831	0.765	1.23	0.320	1.00x	1.00	0.718	0.420	0.286	28.05	1.00x
Naïve	0.05	0.742	0.681	0.454	**4.3**	0.298	1.55x	0.05	0.662	0.326	**0.287**	28.10	1.55x
Ours	0.05	**0.662**	**0.770**	**0.666**	5.1	**0.306**	1.55x	0.05	**0.726**	**0.442**	0.272	**25.23**	1.55x
Naïve	0.10	0.717	0.733	0.565	**3.1**	0.309	1.56x	0.10	0.690	0.370	**0.287**	27.13	1.56x
Ours	0.10	**0.628**	**0.791**	**0.706**	4.1	**0.313**	1.56x	0.10	**0.736**	**0.461**	0.272	**24.28**	1.56x
Naïve	0.20	0.680	0.772	0.645	**2.5**	**0.316**	1.43x	0.20	0.704	0.399	**0.287**	27.00	1.43x
Ours	0.20	**0.576**	**0.809**	**0.740**	3.0	0.316	1.43x	0.20	**0.743**	**0.464**	0.272	**24.79**	1.43x

Table 9: Flux.2-Klein-4B (Labs, 2025) model Evaluation of generation quality with ***Sparse Context***.

	Instruction Driven Image Editing							Reference Based Personalization					
	Tok. f	LPIPS ↓	CLIP-I ↑	DINO ↑	KID ↓ ($\times 10^{-3}$)	CLIP T-I ↑	Speedup ↑	Tok. f	CLIP-I ↑	DINO ↑	CLIP T-I ↑	KID ↓ ($\times 10^{-3}$)	Speedup ↑
Baseline	1.00	0.430	0.826	0.772	1.5	0.317	1.00x	1.00	0.729	0.428	0.281	25.0	1.00x
Random	0.05	0.766	0.640	0.354	12.6	0.286	2.07x	0.05	0.655	0.298	0.283	26.8	2.07x
Ours	0.05	**0.646**	**0.740**	**0.597**	**2.8**	**0.307**	2.07x	0.05	**0.726**	**0.394**	**0.284**	**23.9**	2.07x
Random	0.10	0.736	0.662	0.413	9.7	0.290	1.61x	0.10	0.685	0.343	0.283	26.2	1.61x
Ours	0.10	**0.597**	**0.774**	**0.676**	**2.4**	**0.315**	1.61x	0.10	**0.744**	**0.423**	**0.283**	**23.4**	1.61x
Random	0.20	0.692	0.711	0.529	5.3	0.301	1.43x	0.20	0.707	0.383	**0.283**	26.9	1.43x
Ours	0.20	**0.546**	**0.799**	**0.730**	**2.0**	**0.319**	1.43x	0.20	**0.753**	**0.442**	0.281	**22.3**	1.43x

A.5 Token dropping ablation with different fractions of reference tokens

We ablate over different fractions of reference tokens in Fig. 9 and Tab. 10 where our method achieves good performance even with a small number of reference tokens but gradually improves with higher token keep fractions. Notably, we trained with keep fraction $f \in (0.5, 0.20)$; it can be adapted to a larger keep fraction during inference. Results for Instruction-Driven Image Editing are shown in Fig. 9.

Table 10: Evaluation of generation with different reference token fractions f.

	Instruction Driven Image Editing							Reference Based Personalization					
	Tok. f	LPIPS ↓	CLIP-I ↑	DINO ↑	KID ↓ ($\times 10^{-3}$)	CLIP T-I ↑	Speedup ↑	Tok. f	CLIP-I ↑	DINO ↑	CLIP T-I ↑	KID ↓ ($\times 10^{-3}$)	Speedup ↑
Baseline	1.00	0.433	0.833	0.760	1.5	0.315	1.00x	1.00	0.750	0.451	0.290	21.2	1.00x
Naïve	0.05	0.740	0.675	0.446	**4.5**	0.297	2.06x	0.05	0.654	0.318	**0.286**	28.6	2.06x
Ours	0.05	**0.665**	**0.772**	**0.660**	5.2	**0.304**	2.06x	0.05	**0.735**	**0.435**	0.275	**23.7**	2.06x
Naïve	0.10	0.719	0.723	0.553	3.5	0.309	1.83x	0.10	0.681	0.360	**0.287**	27.8	1.83x
Ours	0.10	**0.628**	**0.796**	**0.703**	**3.4**	**0.311**	1.83x	0.10	**0.748**	**0.452**	0.274	**22.7**	1.83x
Naïve	0.20	0.687	0.762	0.631	3.0	**0.316**	1.49x	0.20	0.692	0.390	**0.288**	27.1	1.49x
Ours	0.20	**0.576**	**0.817**	**0.738**	**2.4**	0.314	1.49x	0.20	**0.754**	**0.455**	0.274	**23.0**	1.49x
Naïve	0.30	0.655	0.783	0.675	2.4	**0.319**	1.36x	0.30	0.699	0.398	**0.287**	28.4	1.36x
Ours	0.30	**0.532**	**0.825**	**0.762**	**2.1**	0.316	1.36x	0.30	**0.755**	**0.457**	0.273	**23.4**	1.36x
Naïve	0.50	0.571	0.807	0.724	1.9	**0.319**	1.27x	0.50	0.705	0.405	**0.287**	28.3	1.27x
Ours	0.50	**0.464**	**0.837**	**0.788**	**1.6**	0.317	1.27x	0.50	**0.756**	**0.457**	0.273	**24.1**	1.27x
Naïve	0.75	0.489	0.819	0.750	1.5	**0.319**	1.11x	0.75	0.709	0.412	**0.287**	28.0	1.11x
Ours	0.75	**0.426**	**0.842**	**0.794**	**1.4**	0.318	1.11x	0.75	**0.755**	**0.454**	0.273	**26.1**	1.11x

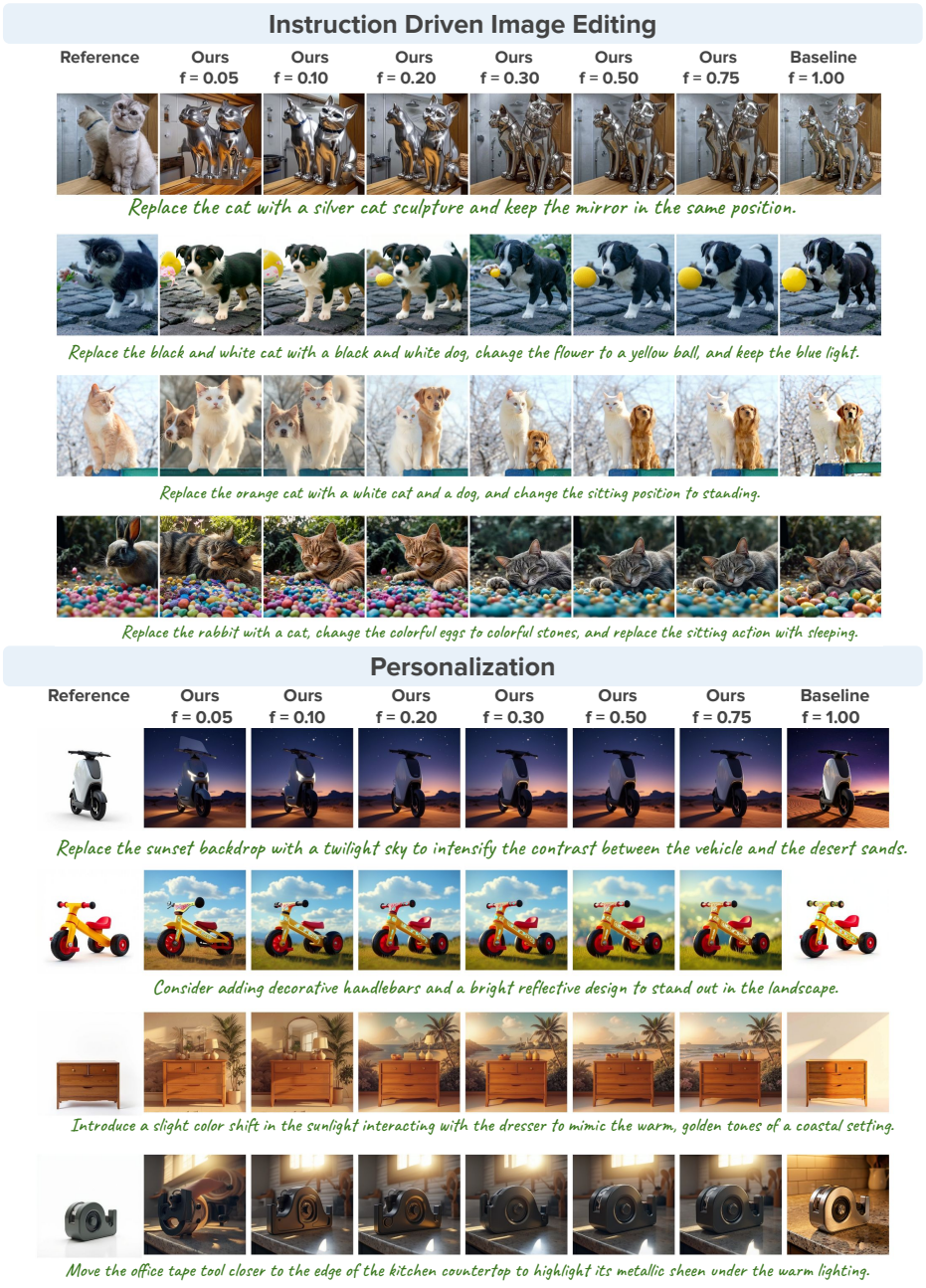


Figure 9: Instruction-Driven Image Editing and Personalization over different drop fractions

A.6 QUALITATIVE COMPARISON WITH BASELINES

We present qualitative results for baseline method comparison of both tasks in Fig. 10.



Figure 10: Comparison with an existing inference-time token reduction method. Our method preserves the object identity well while generating consistent edits.

A.7 Qualitative results for reference conditioned video model

The accompanying `video_results.zip` contains an `index.html` file providing an interactive visualization of the video results. Figure 11 presents qualitative results for reference-conditioned video generation across Canny, depth, and grayscale conditioning on Minimax-H3 MiniMax (2026). Our method maintains visual consistency with the baseline while using substantially fewer reference tokens. In particular, as the reference token fraction is reduced, our method preserves the structure and appearance of the generated videos more effectively than the corresponding naive baseline, demonstrating the effectiveness of our token reduction strategy.

A.8 Robustness to Saliency Map Errors

Our method does not require precise per-pixel saliency values; it only needs an approximate localization of the subject. Since token selection is performed at the coarse 32×32 latent resolution, small errors in the predicted saliency map are naturally averaged out when the map is downsampled to the token grid. To quantify this sensitivity, we perturb the predicted saliency maps with random scaling and pixel shifts before token selection, and evaluate the resulting personalization quality at a 20% token ratio (Tab. 11).

Table 11: **Saliency perturbation** analysis on personalization at 20% tokens.

	Tok. f	CLIP-I ↑	DINO ↑	CLIP T-I ↑	KID ↓ ($\times 10^{-3}$)
Baseline	1.00	0.750	0.451	0.290	21.23
Naïve	0.20	0.692	0.390	**0.288**	27.10
Ours	0.20	**0.754**	**0.455**	0.274	**23.01**
Ours + Sal. Perturb.	0.20	0.752	**0.455**	0.274	23.08

As shown in Tab. 11, the perturbed setting closely matches our unperturbed results across all metrics. Identity preservation is essentially unchanged (CLIP-I 0.752 vs. 0.754; DINO 0.455 for both), text alignment is identical (CLIP T-I 0.274), and KID changes only marginally (23.08 vs. 23.01). The perturbed variant also continues to outperform the naïve token reduction baseline by a large margin in identity preservation and KID. These results confirm that our method is robust to errors in the saliency predictor.

A.9 Additional Qualitative results

We present additional qualitative results for ***Sparse Context*** against the Naïve baseline for instruction-driven image editing in Fig. 12, for single-reference personalization in Fig. 13, and for multi-reference personalization in Fig. 14.

A.10 Implementation Details.

We use FLUX.2 Klien-9B Labs (2025) model for all our experiments. We used 28 step schedule for training and inference of the model. The training took 20 hrs on two A100 GPUS. The learning rate was $1e-4$ with a constant learning warmup and the training use 4 steps of gradient accumulation. The LoRA rank is 16 and is learned only over the attention layer parameters. For inference tasks, we use first apply a Gaussian filter of 3×3 on the input image and then extract Canny edge map for image editing application and use off-the-shelf saliency prediction network Kümmerer et al. (2025) to obtain a single channel saliency map for the reference images. We additionally present the true runtime on A100 machine in the Fig. 15

A.11 Dataset Generation

Instruction-driven editing dataset. Training. For training, we use a subset of the HQ-Edit dataset. We use the source images and edit instructions from HQ-Edit and generate the corresponding edited images using the Flux-Klein-9B model to obtain accurate ground-truth edits, as the targets provided in the original dataset are not accurate.

Evaluation. We evaluate instruction-driven image editing on the PIE-Bench dataset. Since PIE-Bench provides source and target prompt pairs for each edit, we use Qwen3-8B to automatically generate natural language editing instructions from each pair, yielding a set of (`source image`, `edit instruction`) pairs used throughout evaluation.

Single reference personalization. For both training and testing of single reference personalization, we use a subset of the Subject-200K dataset consisting of simple objects in different compositions. We selected a $30K$ dataset for training and another 500 samples for evaluation.

Video Dataset. For video-conditioned editing, we use a subset of the OpenVid-1M (Nan et al., 2025) dataset. We randomly sample 3K video-text pairs from the dataset and use the original videos as ground-truth targets. For each video, we generate three conditioning modalities: Canny edge maps, grayscale maps, and depth maps using Depth Anything V2 (Yang et al., 2024). This results in 9K training pairs, each consisting of a $(\texttt{condition}, \texttt{text prompt}, \texttt{ground-truth video})$ triplet.

Multi-reference personalization dataset. For multi-subject personalization, we construct a synthetic compositional dataset using objects from the Custom Diffusion Objects dataset, grouping object instances into semantic categories solely for structured sampling. We generate approximately 100 prompt templates covering varying object counts and spatial relations, e.g., *"Place $\{a\}$ on top of $\{b\}$ while $\{c\}$ is placed beside $\{d\}$"*, where placeholders are populated using randomly sampled object instances from the categorized groups. For each example, the number of objects is drawn from a categorical distribution over $\{2, 3, 4, 5, 6\}$ with probabilities $[0.3, 0.3, 0.2, 0.1, 0.1]$, respectively. A prompt template corresponding to the sampled object count is then selected and instantiated accordingly. This procedure yields a diverse set of multi-object compositions with varied spatial arrangements and interaction patterns. For testing, we generate 200 test scenes generated through the same procedure as discussed above.

A.12 Memory overhead

We benchmark the additional dynamic memory overhead during the inference process in Fig. 16. As we scale the number of reference images, our method significantly reduced memory consumptions as compared to the baseline without token dropping. This experiment was performed on RTX-6000 Pro with 96G to fit the larger image resolution.

A.13 Limitation

Though ***Sparse Context*** enables inference time control over selecting a best suited approach selecting tokens, the selection process may depend on the input image. Consider the scenario given in Fig. 17 about image editing, where we show two different editing scenarios. We use canny edge based token sampling for editing. In example A, the edit is stylization and using Canny edge preserves the girls identity well as compared to random selection. However in example B, the edit instruction is to replace the object, and using canny based sampling can hurt the preservation of background region as the structure boundary of the bird is dominant.



Figure 11: **Results for reference-conditioned video generation.** Given the reference conditioning video (e.g. canny) our model generates consistent output to the baseline while using only a small set of reference tokens.

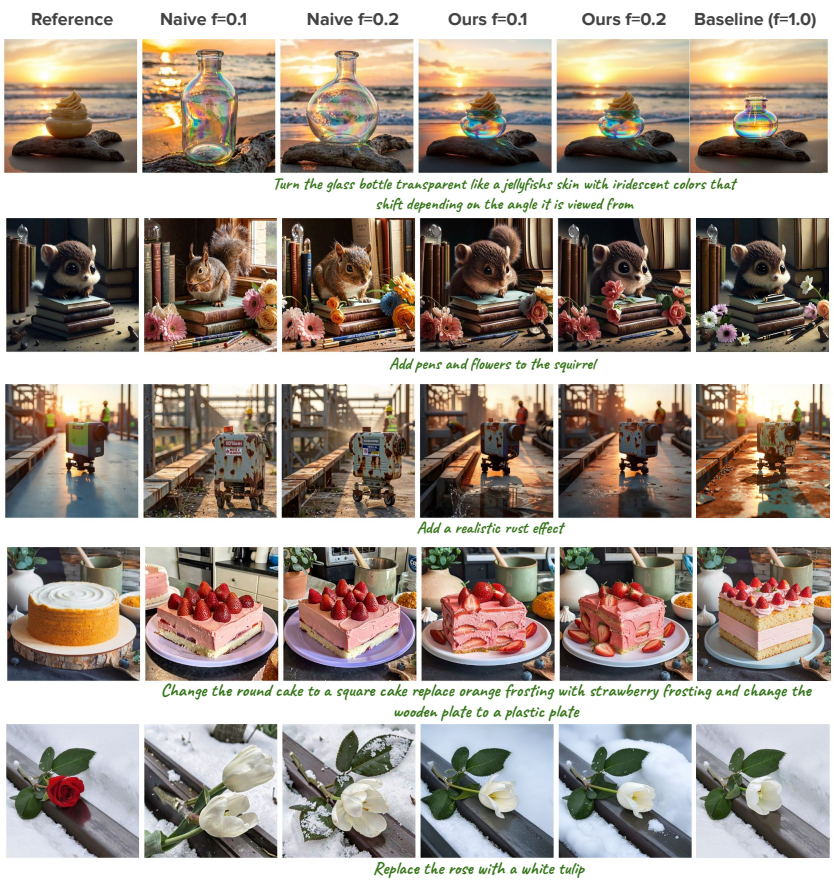


Figure 12: Results for efficient instruction-based image editing.

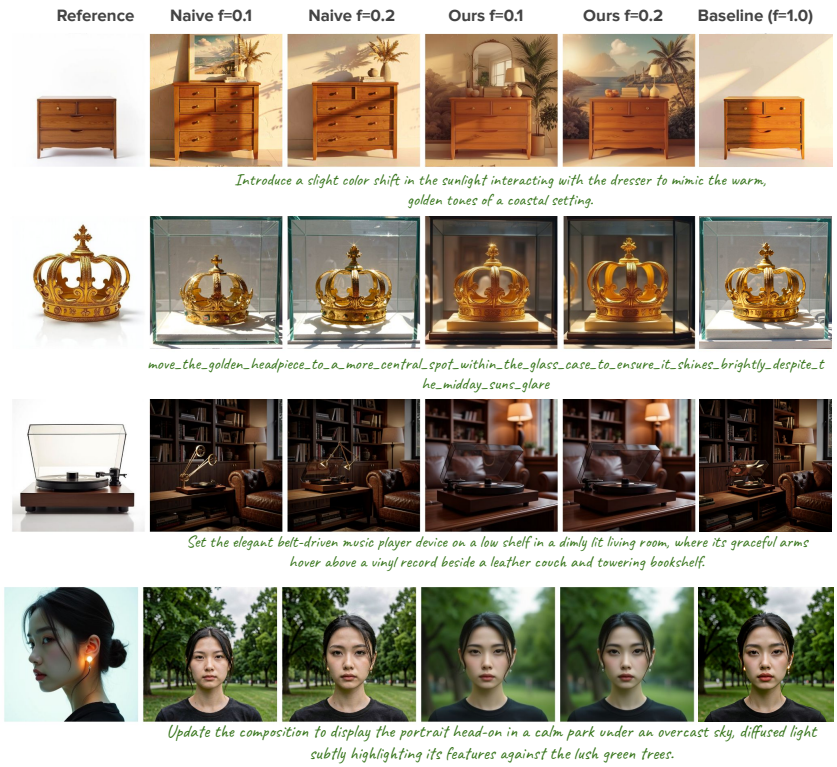


Figure 13: Results for single-reference personalization with ***Sparse Context***.

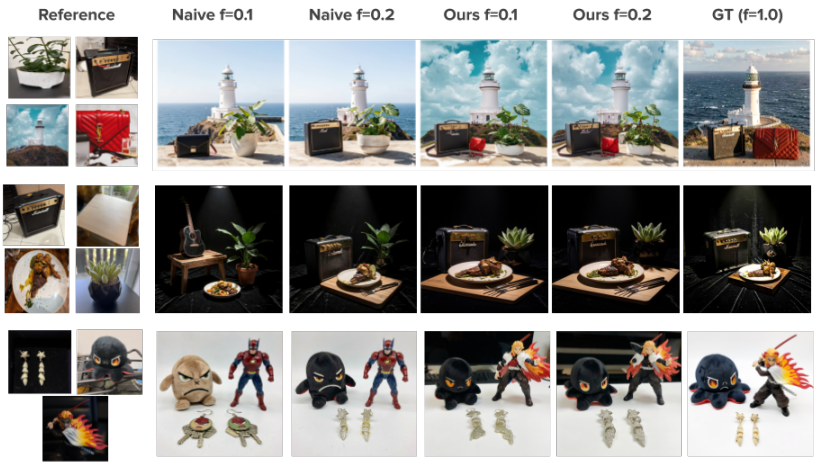


Figure 14: Results for multi-reference personalization with ***Sparse Context***. We compare the Naïve baseline and ***Sparse Context*** at token drop fractions $f = 0.1$ and $f = 0.2$.

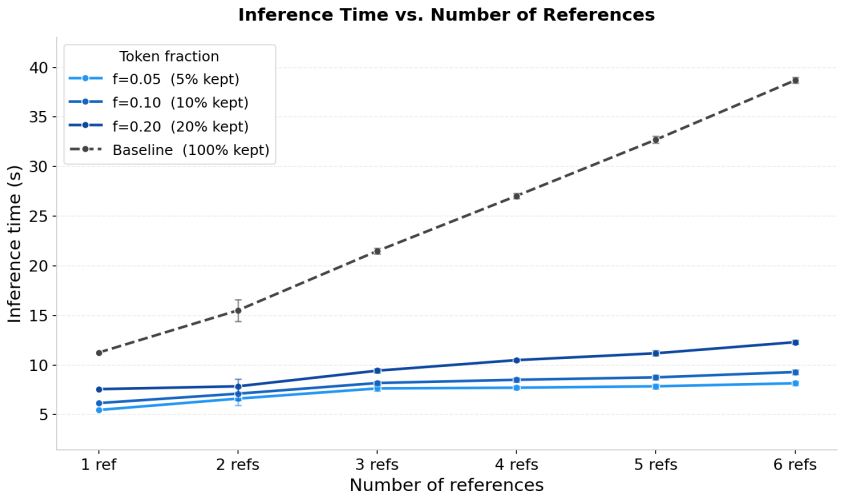


Figure 15: Clock time for inference on A100 GPU

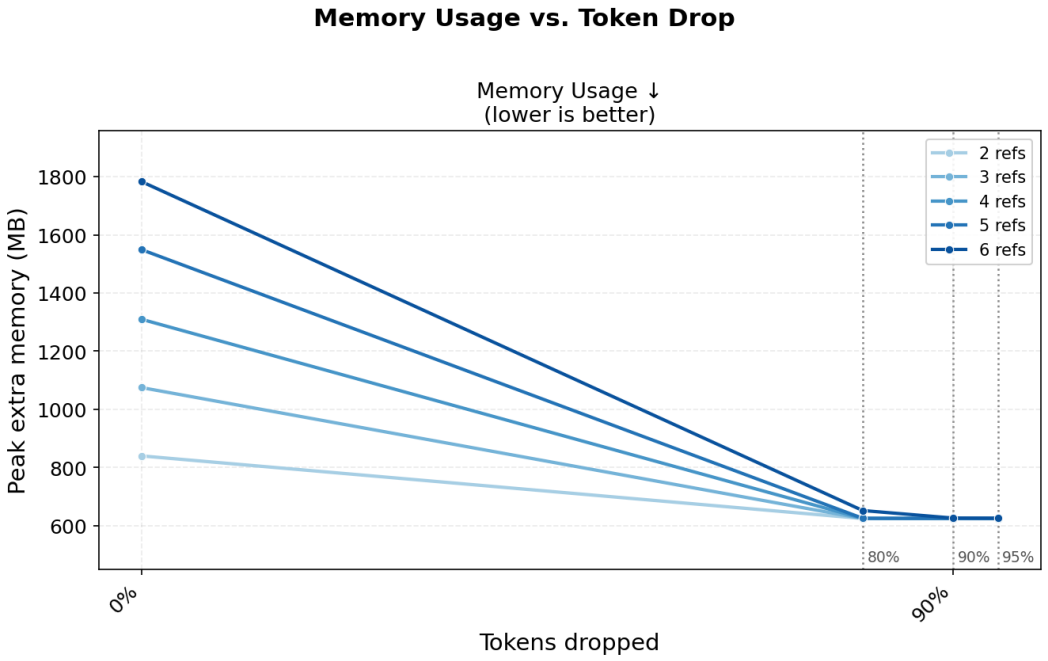


Figure 16: Peak memory during inference with number of reference image and differen token dropping fractions

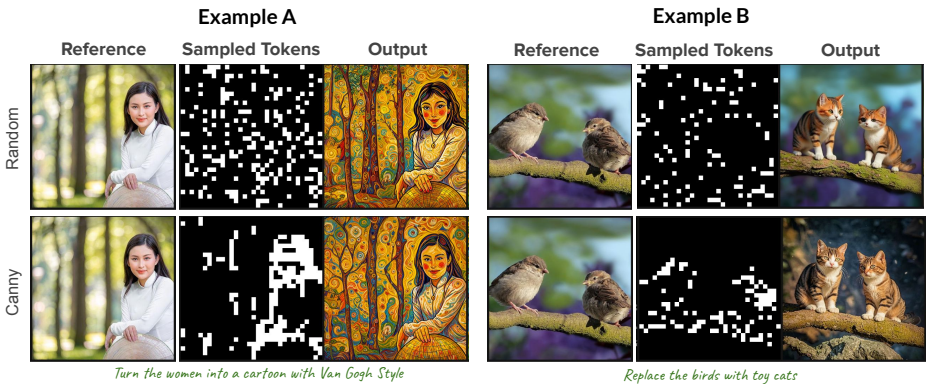


Figure 17: **Limitation.** The choice of the inference time token selection heuristic can depend on the type of edit. In example A stylization edit Canny based sampling is effective but for example B, it will focus only the object region that needs to be replaced, resulting in inferior quality of the background region.