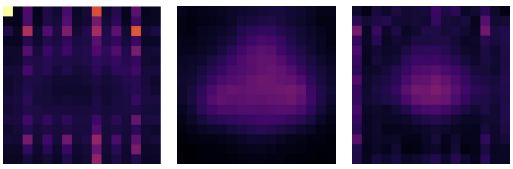
MASKALIGN: TOKEN-SUBSET REPRESENTATION ALIGNMENT FOR EFFICIENT DIFFUSION TRAINING

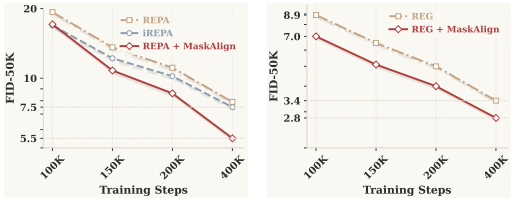
Anonymous authors
Paper under double-blind review

ABSTRACT

Representation alignment with pretrained vision models has recently shown strong potential for accelerating diffusion transformer training. By aligning intermediate diffusion features with representations of clean images from pretrained vision encoders, existing methods improve convergence and generation quality. Recent studies further suggest that the effectiveness of representation alignment is closely related to spatial structure. In this paper, we study representation alignment across different samples. We find that, when representation alignment is applied to all tokens, tokens with large alignment gradient norms exhibit a stable spatial preference across different samples, suggesting that the alignment objective does not affect all tokens uniformly and may encourage the model to fit spatial patterns associated with the encoder rather than visual features specific to each image, which may interfere with the learning of representations that reflect image content. To address this issue, we propose MaskAlign, a representation alignment method that applies alignment to randomly sampled token subsets during training. By exposing the model to different token subsets across iterations, MaskAlign reduces the repeated reinforcement of stable spatial preferences and encourages the model to learn features that are more closely related to individual images. To mitigate the information loss caused by directly dropping tokens, we further introduce a lightweight token mixing block that shares information across tokens before masking. Experiments on ImageNet 256×256 show that MaskAlign consistently improves training convergence and generation quality. On SiT-XL/2, applying MaskAlign to REPA reduces FID from 7.9 to 5.5 at 400K iterations and from 6.4 to 4.9 at 1M iterations. When applied to REG, MaskAlign reduces FID from 3.4 to 2.8 at 400K iterations and from 2.7 to 2.4 at 1M iterations, while reducing training time per step by 11.7%. For REG, the improvement also holds when FID is computed using DINOv2-L, CLIP-L, and VGG16 features, both without and with classifier-free guidance.



(a) Alignment heatmaps using all tokens (DINOv2-B, DINOv3-B, CLIP-B)



(b) FID

Figure 1: (a) Alignment heatmaps using all tokens with DINOv2-B, DINOv3-B, and CLIP-B. Across different encoders, representation alignment over all tokens consistently exhibits stable spatial preferences, suggesting that the learned alignment can depend on spatial patterns associated with the encoder rather than visual content specific to each image. Such preferences may encourage the model to rely on specific spatial positions and interfere with the learning of features that depend on image content. (b) MaskAlign reduces the repeated reinforcement of stable spatial preferences and encourages the model to learn features that are more closely related to individual images. Our simple approach consistently improves the convergence speed of REPA and REG across diverse settings.

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.

1 INTRODUCTION

Diffusion models have advanced significantly in recent years (Ho et al., 2020; Song et al., 2020; Nichol et al., 2021; Balaji et al., 2022; Rombach et al., 2022; Black Forest Labs, 2024; Saharia et al., 2022). Latent diffusion models (LDMs) (Rombach et al., 2022) utilize a Variational Autoencoder (VAE) (Kingma & Welling, 2013) to shift the image generation process from the pixel space to the latent space. DiT (Peebles & Xie, 2023) improves scalability through a Transformer architecture, and SiT (Ma et al., 2024) further enhances performance by employing stochastic interpolants defined in continuous time. Despite these advances, training high-quality image generation models at scale remains computationally expensive.

Pretrained vision models have recently been used to accelerate diffusion training by providing visual representations as additional supervision. REPA (Yu et al., 2025) directly aligns intermediate diffusion features with those of a vision encoder to improve convergence and generation quality. Subsequent work extends this framework using class tokens (Wu et al., 2025), shared latent feature coupling (Kouzelis et al., 2025), representation alignment at the VAE level (Leng et al., 2025), and other objectives based on representation alignment (Wang et al., 2025; Petsangourakis et al., 2025; Singh et al., 2025).

Recent studies further suggest that spatial structure plays an important role in the effectiveness of representation alignment (Singh et al., 2025; Xu et al., 2026). These studies mainly investigate how spatial information is represented or transferred within individual samples, while paying less attention to the spatial behavior of alignment across different samples. In particular, if the spatial supervision introduced by representation alignment consistently favors certain positions across images, the model may repeatedly capture spatial patterns associated with the reference encoder rather than visual features specific to each image.

We investigate this question by studying how the gradient of the alignment loss is distributed across tokens. As shown in Figure 1a, certain spatial positions appear substantially more often among the top 10% of tokens ranked by gradient norm, even after averaging over many images. Moreover, similar stable spatial preferences are observed with different vision encoders, while their spatial patterns vary across encoders. These results suggest that representation alignment over all tokens does not affect all spatial positions uniformly and may introduce persistent spatial patterns related to the reference encoder rather than the visual content of individual images.

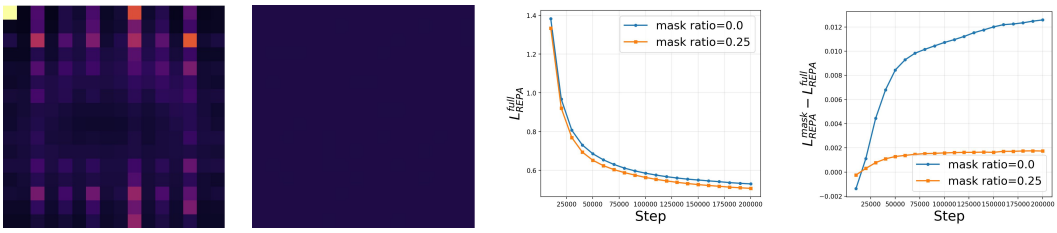
Such persistent spatial preferences may interfere with the learning of features specific to each image. Since alignment over all tokens repeatedly supervises all spatial positions at every iteration, the emphasis on the same spatial positions can be reinforced across samples, encouraging the model to rely on these recurring spatial patterns during representation learning.

Building on these observations, we adopt a dropout-like strategy inspired by random feature dropping for preventing co-adaptation (Baldi & Sadowski, 2013; Wager et al., 2013): instead of aligning all patch tokens at every iteration, we randomly sample a subset of tokens for representation alignment.

By varying the aligned token subset across iterations, MaskAlign reduces the repeated reinforcement of the same spatial positions and encourages the model to learn representations that rely less on persistent spatial preferences. At the same time, directly dropping tokens may disrupt fine-grained spatial information. We therefore introduce a lightweight token mixing block that shares information across tokens before masking, allowing the retained tokens to preserve contextual information from the full feature map.

Figures 2c and 2d show that masked training not only reduces the alignment loss, but also narrows the gap in alignment loss between randomly masked inputs and inputs containing all tokens. This indicates that the learned alignment behavior becomes less sensitive to which subset of tokens is retained and less dependent on particular spatial configurations. Figure 1b further reports FID over training steps on ImageNet 256×256 (Deng et al., 2009). MaskAlign consistently improves both REPA and REG across different training budgets. On SiT-XL/2, it improves REPA from 7.9 to 5.5 FID and REG from 3.4 to 2.8 FID at 400K iterations. Together with the lower cost per step introduced by token masking, these results show that MaskAlign improves both convergence and training efficiency.

In summary, our contributions are as follows:



(a) Full-token heatmap **(b)** 25% mask heatmap **(c)** Alignment loss **(d)** Alignment loss gap

Figure 2: Behavior across tokens and alignment stability under token masking. **(a,b)** Heatmaps show the probability that each spatial position appears among the tokens in the top 10% by alignment gradient, using the same color range $[0, 0.8]$. For reference, a uniform distribution would correspond to approximately 10% for each position. **(a)** Alignment over all tokens exhibits a stable spatial preference. **(b)** A 25% mask ratio substantially reduces this concentrated pattern. **(c)** MaskAlign lowers the alignment loss computed over all tokens. **(d)** MaskAlign narrows the gap in alignment loss $L_{\text{REPA}}^{\text{mask}} - L_{\text{REPA}}^{\text{full}}$, indicating improved stability under perturbations to token subsets

- We analyze the behavior of representation alignment across samples and tokens. We find that, when representation alignment is applied to all tokens, tokens with large alignment gradients exhibit stable spatial preferences across different samples. These preferences persist across multiple vision encoders while varying with the choice of encoder, suggesting that representation alignment may repeatedly emphasize spatial patterns associated with the encoder rather than visual features specific to each image.
- We propose MaskAlign, a random token masking strategy that applies alignment to randomly sampled token subsets instead of all patch tokens at every iteration. Motivated by dropout's ability to prevent co-adaptation, MaskAlign disrupts the repeated reinforcement of specific spatial positions and encourages representation learning that is less dependent on persistent spatial preferences. We further introduce a lightweight token mixer to preserve contextual information before masking.
- We validate the effectiveness of MaskAlign on ImageNet 256×256. MaskAlign consistently improves both REPA and REG across different training budgets. On SiT-XL/2, it improves REPA from 7.9 to 5.5 FID and REG from 3.4 to 2.8 FID at 400K iterations. The improvements remain consistent when FID is computed using DINOv2-L, CLIP-L, and VGG16 features, both without and with classifier-free guidance. MaskAlign also reduces the alignment loss measured over all tokens and improves alignment stability under random token masking.

2 Related Work

Generative Models for Image Generation. Early diffusion models, such as DDPM (Ho et al., 2020) and DDIM (Song et al., 2020), perform denoising in the pixel space. Latent Diffusion Models (LDMs) (Rombach et al., 2022) improve efficiency by performing the generative process in a compact latent space obtained with a VAE (Kingma & Welling, 2013). DiT (Peebles & Xie, 2023) further improves scalability with a Transformer architecture, while SiT (Ma et al., 2024) extends this framework using continuous-time stochastic interpolants. Despite these advances, training high-quality diffusion models still requires substantial computational resources.

Efficient Training via Token Masking. Token masking is a common approach for accelerating diffusion training. MDT (Gao et al., 2023) and MaskDiT (Zheng et al., 2023) reduce the number of input tokens during training and reconstruct the full token set from visible tokens. MicroDiT (Sehwag et al., 2025) introduces a lightweight mixer before masking to preserve token information, while TREAD (Krause et al., 2025) routes a subset of tokens across intermediate layers to reduce computation. In contrast, MaskAlign uses random masking to construct token subsets for representation alignment rather than to reconstruct missing tokens. The prediction and alignment losses are computed on the visible patch tokens, with the class token preserved when using REG.

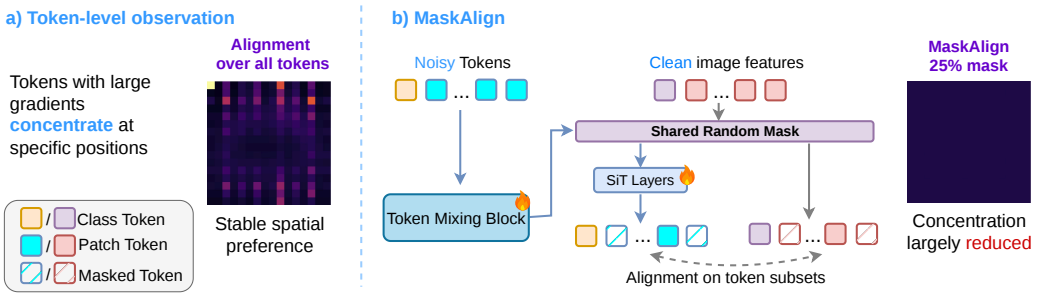


Figure 3: Overview of MaskAlign. **a)** Alignment over all tokens exhibits a stable spatial preference, with tokens that have large alignment gradients concentrated at specific spatial positions. **b)** MaskAlign first mixes information across tokens and then uses a shared random mask to compute representation alignment over token subsets while preserving the class token. With a 25% mask ratio, MaskAlign substantially reduces the concentrated spatial pattern.

Representation Alignment with External Models. REPA (Yu et al., 2025) aligns intermediate DiT/SiT features with representations from a pretrained vision model to improve convergence and generation quality. REG (Wu et al., 2025) and ReDi (Kouzelis et al., 2025) further improve the transfer of semantic information, while REPA-E (Leng et al., 2025) applies representation alignment to VAE training. HASTE (Wang et al., 2025) shows that continued alignment at later training stages can conflict with the generation objective. iREPA (Singh et al., 2025) shows that spatial structure is important for representation alignment, and sREPA (Xu et al., 2026) further aligns the relational structure among spatial tokens. These methods mainly study spatial structure within individual samples, whereas MaskAlign focuses on the spatial behavior of alignment across different samples. PixelREPA (Shin et al., 2026) introduces partial token masking in a Transformer adapter to constrain representation alignment for pixel-space diffusion transformers. In contrast, MaskAlign studies persistent spatial preferences across samples in latent diffusion models and uses random token-subset alignment to reduce their repeated reinforcement.

3 Preliminaries

Scalable Interpolant Transformers (SiT). Our method follows the SiT framework (Ma et al., 2024), which is based on stochastic interpolants (Lipman et al., 2022). Given a clean image z_*, a pretrained VAE encoder $\mathcal{E}_z$ maps it to a latent representation z_0. At timestep t, SiT constructs the noisy latent as $z_t = \alpha_t z_0 + \sigma_t \epsilon_z$, where $\epsilon_z \sim \mathcal{N}(0, I)$, and trains a Transformer $v_\theta(z_t, t)$ to predict the target velocity $\dot{\alpha}_t z_0 + \dot{\sigma}_t \epsilon_z$ using a squared error objective. Following the standard SiT setting, we use $\alpha_t = 1 - t$ and $\sigma_t = t$, with $\dot{\alpha}_t = -1$ and $\dot{\sigma}_t = 1$.

4 Token-level Analysis

4.1 Distribution of Alignment Gradients

Recent studies (Singh et al., 2025; Xu et al., 2026) have shown that spatial structure plays an important role in representation alignment. Existing analyses, however, mainly focus on how spatial information is represented and transferred within individual samples. In this section, we instead examine whether representation alignment exhibits consistent spatial preferences across different samples. We first consider the setting in which all patch tokens are aligned with their corresponding reference features. Since the class token has no spatial position, our heatmap analysis focuses on patch tokens. Given the hidden state $h_i^{[\ell_a]}$ at layer ℓ_a and the reference feature r_i, the alignment loss is defined as

$$\mathcal{L}_{\text{REPA}} = -\frac{1}{N} \sum_{i=1}^{N} \text{sim}\left(r_i, h_\phi(h_i^{[\ell_a]})\right), \tag{1}$$

where $h_\phi(\cdot)$ is the alignment projector. We omit the expectation over samples and timesteps for simplicity.

To examine how this objective affects training, we analyze the gradient norms of $\mathcal{L}_{\mathrm{REPA}}$ with respect to the hidden states at layer ℓ_a. We focus on this layer because it is where the alignment supervision is explicitly injected through the projector.

Let $h_i^{[\ell_a]}$ denote the hidden state of the i-th patch token at the alignment layer ℓ_a. We compute the norm of the alignment gradient for each patch token as

$$g_i^{\mathrm{align}} = \left\| \frac{\partial \mathcal{L}_{\mathrm{REPA}}}{\partial h_i^{[\ell_a]}} \right\|_2 . \tag{2}$$

For each image, we select the top-k patch tokens with the largest gradient norms. We then compute the probability that each spatial position appears in this top-k set across multiple images. We set k to 10% of the patch tokens.

As shown in Figure 2a, certain spatial positions remain more likely to appear in the top-k set, even after averaging over many images. The largest spatial probability is about $21\times$ the smallest, suggesting that this preference cannot be explained by minor random fluctuations. Figure 1a shows that similar stable spatial preferences are observed with different vision encoders, while their spatial patterns vary across encoders.

These results reveal a stable spatial concentration of tokens with large alignment gradients across samples. Moreover, different vision encoders produce different spatial patterns under the same setting in which all tokens are aligned. This suggests that these stable preferences may be related to the reference encoder rather than the visual content of individual images.

As a result, the alignment objective may repeatedly emphasize the same spatial positions during training. Consequently, the model may learn features associated with these recurring spatial patterns at the expense of features that reflect image content.

4.2 Motivation for Alignment over Token Subsets

The observation above suggests that alignment over all tokens can repeatedly reinforce stable spatial preferences across different samples. Since these preferences may be related to the reference encoder rather than the content of individual images, repeatedly aligning all spatial positions may encourage the model to rely on the same preferred positions during training.

Building on this observation, MaskAlign applies random token masking during representation alignment. Inspired by random feature dropping, which is commonly used to reduce co-adaptation (Baldi & Sadowski, 2013; Wager et al., 2013), we randomly sample subsets of patch tokens during training. As the aligned token subsets vary across iterations, the same spatial positions are less consistently reinforced. This reduces the model's reliance on persistent spatial preferences and encourages it to learn features that are more closely related to individual images.

5 MaskAlign

5.1 Framework

The overall framework of MaskAlign is shown in Figure 3. During training, representation alignment is applied to a randomly sampled subset of patch tokens. For REG, MaskAlign additionally prepends a class token with global semantics to the patch tokens. When applying MaskAlign to REPA, we omit the class token and its prediction loss, while applying the same random masking and alignment procedure to patch tokens. Before masking, we apply a lightweight token mixing block to share information across tokens and mitigate the disruption from dropping patch tokens. The mixed class token and visible patch tokens are then fed into the diffusion transformer. Random token masking is used only during training; at inference time, all tokens are retained.

Token Mixing and Random Masking. Following Sec. 3, let z_* denote a clean image and let $z_0 = \mathcal{E}_z(z_*) \in \mathbb{R}^{D_z \times H_z \times W_z}$ be its clean latent. At timestep t, the noisy latent is constructed as

$$z_t = \alpha_t z_0 + \sigma_t \epsilon_z, \quad \epsilon_z \sim \mathcal{N}(0, I). \tag{3}$$

The noisy latent z_t is patchified and projected into a sequence of patch tokens $x_t^0 = \{x_{t,1}^0, \ldots, x_{t,N}^0\} \in \mathbb{R}^{N \times D}$, where N is the number of patch tokens and D is the hidden dimension. Following REG, we prepend a class token c_t^0 to form $H_t^0 = [c_t^0, x_t^0] \in \mathbb{R}^{(N+1) \times D}$.

Before random token masking, we apply a lightweight token mixing block $M_\psi(\cdot)$ to share information across tokens:

$$\bar{H}_t^0 = [\bar{c}_t^0, \bar{x}_t^0] = M_\psi(H_t^0, t, y), \tag{4}$$

where y denotes the class condition. This step mitigates the disruption caused by directly dropping patch tokens.

In our implementation, M_ψ consists of two standard SiT blocks with the same architecture and conditioning mechanism as the backbone blocks. The mixing blocks follow the same initialization scheme as the SiT backbone. Unless otherwise specified, we use a mask ratio of 25%. During inference, the mixing blocks remain active, but random masking is disabled and all patch tokens are retained.

We then sample a binary keep mask $m \in \{0,1\}^N$ independently for each training sample, where every patch token has the same probability of being masked. Let $S(m) = \{i \mid m_i = 1\}$ denote the indices of the visible patch tokens, with $N_m = |S(m)|$. The class token is always preserved, while random masking is applied only to patch tokens:

$$\widetilde{H}_t^0(m) = \left[\bar{c}_t^0, \{\bar{x}_{t,i}^0\}_{i \in S(m)}\right] \in \mathbb{R}^{(1+N_m) \times D}, \tag{5}$$

where $[\,\cdot\,]$ denotes sequence concatenation. The masked sequence $\widetilde{H}_t^0(m)$ is then fed into the following SiT blocks. At layer ℓ, the transformer produces

$$H_t^{[\ell]}(m) = \left[h_{t,\mathrm{cls}}^{[\ell]}(m), \{h_{t,i}^{[\ell]}(m)\}_{i \in S(m)}\right]. \tag{6}$$

Training Losses. After random token masking, the prediction loss is computed on the preserved class token and the visible patch tokens. Let $r^* = \{r_{\mathrm{cls}}, r_1, \ldots, r_N\}$ denote the reference features extracted from the clean image by the pretrained vision encoder, where r_{cls} denotes the projected clean class token. Following REG, we construct the noisy class token as $c_t^0 = \alpha_t r_{\mathrm{cls}} + \sigma_t \epsilon_{\mathrm{cls}}$, with target velocity $v_{\mathrm{cls}}^*(t) = \dot{\alpha}_t r_{\mathrm{cls}} + \dot{\sigma}_t \epsilon_{\mathrm{cls}}$. For each visible patch token $i \in S(m)$, let $\hat{v}_i(m,t)$ and $v_i^*(t) = \dot{\alpha}_t z_{0,i} + \dot{\sigma}_t \epsilon_{z,i}$ denote the predicted and target velocities. For the class token, let $\hat{v}_{\mathrm{cls}}(m,t)$ denote the predicted velocity. We weight the prediction loss for the class token by β:

$$\mathcal{L}_{\mathrm{pred}} = \mathbb{E}_{z^*, \epsilon_z, \epsilon_{\mathrm{cls}}, t, m}\left[\frac{1}{N_m} \sum_{i \in S(m)} \|\hat{v}_i(m,t) - v_i^*(t)\|_2^2 + \beta \,\|\hat{v}_{\mathrm{cls}}(m,t) - v_{\mathrm{cls}}^*(t)\|_2^2\right]. \tag{7}$$

At alignment layer ℓ_a, we define the visible alignment index set as $\mathcal{A}(m) = \{\mathrm{cls}\} \cup S(m)$, where the class token is always included. The projector $h_\phi(\cdot)$ maps hidden states into the reference feature space. For each $a \in \mathcal{A}(m)$, let r_a and $h_{t,a}^{[\ell_a]}(m)$ denote the corresponding reference feature and hidden state. The alignment loss is then

$$\mathcal{L}_{\mathrm{REPA}} := -\mathbb{E}_{z_*, \epsilon_z, t, m}\left[\frac{1}{|\mathcal{A}(m)|} \sum_{a \in \mathcal{A}(m)} \mathrm{sim}\Big(r_a, h_\phi(h_{t,a}^{[\ell_a]}(m))\Big)\right], \tag{8}$$

where $|\mathcal{A}(m)| = N_m + 1$. The final training objective is $\mathcal{L}_{\mathrm{total}} = \mathcal{L}_{\mathrm{pred}} + \lambda \mathcal{L}_{\mathrm{REPA}}$, where λ controls the strength of representation alignment.

5.2 Measuring Alignment Stability

To assess alignment stability under random masking, we compare the alignment losses obtained using all tokens and using a randomly sampled subset. Let $\mathcal{L}_{\mathrm{REPA}}^{\mathrm{full}}$ denote the loss computed with the class token and all patch tokens, and $\mathcal{L}_{\mathrm{REPA}}^{\mathrm{mask}}$ denote the loss computed with the class token and a randomly sampled subset of patch tokens. We define their difference as

$$G_r = \mathcal{L}_{\mathrm{REPA}}^{\mathrm{mask}} - \mathcal{L}_{\mathrm{REPA}}^{\mathrm{full}}. \tag{9}$$

Table 1: FID comparison during training on ImageNet 256 × 256 without CFG.

Method	#Params	Iter.	FID↓
SiT-B/2	130M	400K	33.0
REPA	130M	400K	24.4
REG	132M	400K	15.2
REG + **MaskAlign**	154M	400K	**14.8**
SiT-XL/2	675M	7M	8.3
REPA	675M	150K	13.6
iREPA	675M	150K	12.2
REPA + **MaskAlign**	728M	150K	**10.8**
REPA	675M	200K	11.1
iREPA	675M	200K	10.2
REPA + **MaskAlign**	728M	200K	**8.6**
REG	677M	200K	5.0
REG + **MaskAlign**	732M	200K	**4.0**
REPA	675M	400K	7.9
iREPA	675M	400K	7.5
REPA + **MaskAlign**	728M	400K	**5.5**
REG	677M	400K	3.4
REG + **MaskAlign**	732M	400K	**2.8**
REPA	675M	1M	6.4
iREPA	675M	1M	6.7
REPA + **MaskAlign**	728M	1M	**4.9**
REG	677M	1M	2.7
REG + **MaskAlign**	732M	1M	**2.4**

Table 2: Comparison with state-of-the-art methods on ImageNet 256 × 256 with CFG.

Model	Epochs	FID↓	sFID↓	IS↑	Prec.↑	Rec.↑
Autoregressive Models						
VAR	350	1.80	-	365.4	0.83	0.57
MagViTv2	1080	1.78	-	319.4	0.83	0.57
MAR	800	1.55	-	303.7	0.81	0.62
Pixel diffusion						
ADM-U	400	3.94	6.14	186.7	0.82	0.52
VDM++	560	2.40	-	225.3	-	-
Latent Diffusion Models						
LDM	200	3.60	-	247.7	0.87	0.48
U-ViT-H/2	240	2.29	5.68	263.9	0.82	0.57
MaskDiT	1600	2.28	5.67	276.6	0.80	0.61
SD-DiT	480	3.23	-	270.3	0.82	0.59
SiT-XL/2	1400	2.06	4.50	270.3	0.82	0.59
FasterDiT	400	2.03	4.63	264.0	0.81	0.60
MDTV2	1080	1.58	4.52	317.7	0.79	0.65
Leveraging Visual Representations						
REPA	80	1.98	4.60	263.0	0.80	0.61
iREPA	80	1.93	4.59	268.8	0.80	0.60
REPA + **MaskAlign**	80	**1.92**	**4.59**	**270.1**	**0.81**	**0.60**
REG	80	1.86	4.49	321.4	0.76	**0.63**
REG + **MaskAlign**	80	**1.80**	**4.48**	**310.0**	**0.81**	**0.63**
REG	160	1.59	4.36	304.6	0.77	0.65
REG + **MaskAlign**	160	**1.56**	**4.36**	**304.7**	**0.79**	**0.65**

Table 3: **Evaluation with different feature encoders on ImageNet** 256×256**, with and without classifier-free guidance.** FID is computed using DINOv2-L, CLIP-L, and VGG16 features. REG + MaskAlign consistently achieves lower FID than REG across all evaluation encoders and both training budgets, showing that the improvement is not specific to a particular feature space.

		W/o CFG			W/ CFG		
Model	Step	DINOv2-L	CLIP-L	VGG16	DINOv2-L	CLIP-L	VGG16
REG	400K	144.72	22.89	39.86	99.59	17.57	34.06
REG + **MaskAlign**	400K	**114.28**	**18.46**	**34.05**	**83.26**	**15.88**	**31.24**
REG	1M	101.45	18.94	35.61	68.34	13.80	30.10
REG + **MaskAlign**	1M	**86.68**	**16.02**	**32.75**	**63.36**	**13.26**	**28.59**

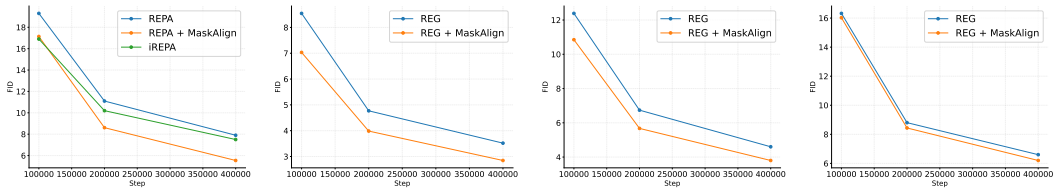
A smaller G_r indicates lower sensitivity to which tokens are retained.

Figures 2c and 2d report the full-token alignment loss and the alignment-loss gap G_r for REG and MaskAlign under a 25% mask ratio. At 200K steps, the gap for MaskAlign is only 13.8% of that for REG, indicating substantially lower sensitivity to random masking. The larger gap for REG suggests that its alignment behavior depends more strongly on the retained token set. These results provide evidence that random token masking encourages more stable alignment behavior under token-subset perturbations.

6 Experiments

6.1 Experimental Setup

Implementation Details. We follow the standard training procedures of SiT, REPA and REG. We conduct experiments on ImageNet, where all images are center-cropped and resized to 256×256 following the ADM (Dhariwal & Nichol, 2021) preprocessing protocol. Each image is then encoded into a latent representation z using the Stable Diffusion VAE. We adopt SiT-B/2 and SiT-XL/2 as the



(a) DINOv2-B **(b)** DINOv2-B **(c)** DINOv3-B **(d)** CLIP-B

Figure 4: (a) Comparison of REPA, iREPA, and REPA + MaskAlign using DINOv2-B. (b-d) Comparison of REG and REG + MaskAlign using DINOv2-B, DINOv3-B, and CLIP-B, respectively. MaskAlign consistently improves the corresponding representation alignment baseline across different encoders, achieving lower FID throughout training and demonstrating its effectiveness across different visual representations.

Table 4: Ablation study on token masking and token mixing. All experiments are conducted on ImageNet 256×256 using SiT-XL/2 models trained for 600K iterations without CFG.

Method	FID↓	sFID↓	IS↑
REG + MaskAlign	**2.67**	**4.78**	**198.10**
w/o Mixing	3.54	6.65	194.51
w/o Masking	3.20	4.92	188.84
w/o Both	3.01	4.88	193.16

Table 5: Computational cost and performance comparison on ImageNet 256×256 at 400K training iterations. Time denotes the average training time per iteration in seconds. Both methods use the SiT-XL/2 backbone and the same GPU hardware.

Method	Params	Time	Tokens	FID ↓
REG	677M	0.359	257	3.4
REG + MaskAlign	732M	0.317	193	**2.8**

backbone architectures. For fair comparison, we use a fixed batch size of 256 and adopt the same learning rate and exponential moving average (EMA) settings as REG.

Evaluation Protocol. We evaluate generation quality using Fréchet Inception Distance (FID) (Heusel et al., 2017), structural FID (sFID) (Nash et al., 2021), Inception Score (IS) (Salimans et al., 2016), precision, and recall (Kynkäänniemi et al., 2019). All metrics are computed using 50K generated images. Following REPA (Yu et al., 2025), we use the SDE Euler-Maruyama solver with 250 sampling steps. More evaluation details are provided in the Appendix.

Accelerating Training Convergence. Table 1 reports the FID scores of different training methods based on representation alignment on ImageNet 256×256 without classifier-free guidance (CFG). Across different backbones and training budgets, MaskAlign improves both REPA and REG at the same number of training iterations.

Figure 1b further compares the convergence curves of REPA (Yu et al., 2025), iREPA (Singh et al., 2025), REG (Wu et al., 2025), and MaskAlign. At 400K iterations, REPA + MaskAlign reduces FID from 7.9 to 5.5, while REG + MaskAlign reduces FID from 3.4 to 2.8. On SiT-XL/2, REG + MaskAlign improves FID from 5.0 to 4.0 at 200K iterations and from 2.7 to 2.4 at 1M iterations. REPA + MaskAlign also improves FID from 13.6 to 10.8 at 150K iterations and from 6.4 to 4.9 at 1M iterations. Notably, REPA + MaskAlign at 400K already outperforms REPA at 1M, while REG + MaskAlign reaches 2.8 FID at 400K, close to the 2.7 FID of REG at 1M. These results show that MaskAlign achieves comparable or better generation quality with substantially fewer training iterations. More experimental comparisons are provided in the Appendix.

Comparison with SOTA Methods. Table 2 compares MaskAlign with recent generative models on ImageNet 256×256 with classifier-free guidance (CFG). At 80 epochs, MaskAlign reduces the FID of REPA from 1.98 to 1.92 and that of REG from 1.86 to 1.80, achieving lower FID than the vanilla SiT-XL/2 trained for 1,400 epochs. At 160 epochs, MaskAlign further improves the FID of REG from 1.59 to 1.56 and increases precision from 0.77 to 0.79. MaskAlign improves both REPA and REG at 80 epochs and continues to improve REG at 160 epochs.

Evaluation with Different Feature Encoders. To test whether the improvement depends on the evaluation feature space, we additionally evaluate REG and REG + MaskAlign using the

dinov2-vit-l-14, clip-vit-l-14, and vgg16 (Simonyan & Zisserman, 2014) feature extractors registered in torch-fidelity (Obukhov et al., 2020). Table 3 reports the results. REG + MaskAlign achieves lower scores with all three feature extractors, both with and without classifier-free guidance, showing that the improvement is not specific to a particular feature space.

Training with Different Vision Encoders. Figure 4 compares the convergence of REPA, iREPA, REPA + MaskAlign, REG, and REG + MaskAlign when trained with DINOv2-B, and further compares REG and REG + MaskAlign using different vision encoders. MaskAlign achieves lower FID at matched training steps and reaches the same FID levels in fewer steps. Moreover, REG + MaskAlign consistently outperforms REG when using DINOv2-B, DINOv3-B, and CLIP-B. Together with the observations in Figure 1, these results show that the spatial preference and the benefit of MaskAlign generalize across different vision encoders.

Computational Cost Comparison. Table 5 compares REG and REG + MaskAlign at 400K training iterations using the same SiT-XL/2 backbone and GPU hardware. Although MaskAlign introduces about 8% more parameters, random token masking reduces the number of input tokens from 257 to 193 and lowers the training time per step from 0.359s to 0.317s. This corresponds to a 24.9% reduction in tokens and an 11.7% reduction in time. Together with the faster convergence shown in Figure 1, this indicates that MaskAlign improves training efficiency in two ways: it reaches the same FID level with fewer iterations and also reduces the training cost per step. REG + MaskAlign improves FID from 3.4 to 2.8, providing better generation quality at a lower training cost per step.

6.2 Ablation

Effect of Token Masking and Token Mixing. We ablate the effects of pre-mask token mixing and random token masking by removing each component separately. As shown in Table 4, the full model achieves the best performance across all metrics, indicating that both components are important for MaskAlign. Removing the token mixing block leads to the worst FID and sFID, suggesting that directly applying random masking without first sharing information across tokens can severely disrupt the input token representations. Meanwhile, removing random masking degrades performance. The variant that uses only token mixing performs worse than the variant without either component. These results show that token mixing and random masking are complementary: pre-mask token mixing reduces the information loss caused by dropping tokens, while random masking exposes the model to different token subsets, which promotes more stable alignment.

Additional Ablations. We further study the effects of the mask ratio and the number of token mixing layers before masking. These experiments are provided in Appendix C. The results show that a moderate mask ratio and a lightweight token mixing block provide the best trade-off between perturbing the token set and preserving information.

7 Conclusion

In this paper, we present MaskAlign, a representation alignment method that operates on token subsets for efficient diffusion transformer training. Building on recent findings that spatial structure plays an important role in representation alignment, we further analyze how alignment supervision behaves across different samples. We observe stable spatial preferences among tokens with large alignment gradients. These preferences vary across vision encoders, suggesting that alignment over all tokens may repeatedly emphasize spatial patterns associated with the reference encoder rather than visual features specific to each image. To reduce the repeated reinforcement of such spatial preferences, MaskAlign applies representation alignment to randomly sampled token subsets. We further use a lightweight token mixing block before masking to reduce the information loss caused by directly dropping tokens. Experiments on ImageNet 256×256 show that MaskAlign improves alignment stability under perturbations to the token set, accelerates training convergence, and achieves better generation quality with lower computational cost per step.

AI USE STATEMENT

We only use generative AI tools to correct grammatical errors and improve the writing of the paper. The research ideas, method design, experimental framework, and main content of the paper are entirely our own work. All AI-assisted text is carefully reviewed and revised to ensure that it accurately describes our method, experiments, and conclusions. We take full responsibility for the accuracy and integrity of the content in this paper.

ETHICS STATEMENT

We use the publicly available ImageNet dataset (Deng et al., 2009) to train and evaluate our models. We follow the applicable terms and licenses of the dataset and other resources used in this work. Our method is designed to improve the training efficiency of image generation models. We support the responsible use of the proposed method with respect for privacy, data licenses, and applicable regulations. We discourage the use of this method for deceptive, harmful, or otherwise inappropriate purposes.

REPRODUCIBILITY STATEMENT

We provide the training and inference code for MaskAlign in the supplementary materials. We describe the proposed method and model architecture in Sec. 5, including the token masking strategy and the alignment procedure. We also report the main training settings and evaluation protocol used in our experiments. These details are provided to facilitate reproduction of our results.

REFERENCES

Yogesh Balaji, Seungjun Nah, Xun Huang, Arash Vahdat, Jiaming Song, Qinsheng Zhang, Karsten Kreis, Miika Aittala, Timo Aila, Samuli Laine, Bryan Catanzaro, Tero Karras, and Ming-Yu Liu. ediff-i: Text-to-image diffusion models with an ensemble of expert denoisers. *arXiv preprint arXiv:2211.01324*, 2022.

Pierre Baldi and Peter J Sadowski. Understanding dropout. *NeurIPS*, 26, 2013.

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

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pp. 248–255. Ieee, 2009.

Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. In *NeurIPS*, 2021.

Shanghua Gao, Pan Zhou, Ming-Ming Cheng, and Shuicheng Yan. Mdtv2: Masked diffusion transformer is a strong image synthesizer. *arXiv preprint arXiv:2303.14389*, 2023.

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. *NeurIPS*, 30, 2017.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *NeurIPS*, 2020.

Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

Theodoros Kouzelis, Efstathios Karypidis, Ioannis Kakogeorgiou, Spyros Gidaris, and Nikos Komodakis. Boosting generative image modeling via joint image-feature synthesis. *arXiv preprint arXiv:2504.16064*, 2025.

Felix Krause, Timy Phan, Ming Gui, Stefan Andreas Baumann, Vincent Tao Hu, and Björn Ommer. Tread: Token routing for efficient architecture-agnostic diffusion training. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 15703–15713, 2025.

Tuomas Kynkäänniemi, Tero Karras, Samuli Laine, Jaakko Lehtinen, and Timo Aila. Improved precision and recall metric for assessing generative models. *NeurIPS*, 32, 2019.

Xingjian Leng, Jaskirat Singh, Yunzhong Hou, Zhenchang Xing, Saining Xie, and Liang Zheng. Repa-e: Unlocking vae for end-to-end tuning of latent diffusion transformers. In *ICCV*, pp. 18262–18272, 2025.

Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. *arXiv preprint arXiv:2210.02747*, 2022.

Nanye Ma, Mark Goldstein, Michael S Albergo, Nicholas M Boffi, Eric Vanden-Eijnden, and Saining Xie. Sit: Exploring flow and diffusion-based generative models with scalable interpolant transformers. In *European Conference on Computer Vision*, pp. 23–40. Springer, 2024.

Charlie Nash, Jacob Menick, Sander Dieleman, and Peter W Battaglia. Generating images with sparse representations. *arXiv preprint arXiv:2103.03841*, 2021.

Alex Nichol, Prafulla Dhariwal, Aditya Ramesh, Pranav Shyam, Pamela Mishkin, Bob McGrew, Ilya Sutskever, and Mark Chen. Glide: Towards photorealistic image generation and editing with text-guided diffusion models. *arXiv preprint arXiv:2112.10741*, 2021.

Anton Obukhov, Maximilian Seitzer, Po-Wei Wu, Semen Zhydenko, Jonathan Kyl, and Elvis Yu-Jing Lin. High-fidelity performance metrics for generative models in pytorch, 2020. URL `https://github.com/toshas/torch-fidelity`. Version: 0.4.0, DOI: 10.5281/zenodo.3786539.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023.

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.

Giorgos Petsangourakis, Christos Sgouropoulos, Bill Psomas, Theodoros Giannakopoulos, Giorgos Sfikas, and Ioannis Kakogeorgiou. Reglue your latents with global and local semantics for entangled diffusion. *arXiv preprint arXiv:2512.16636*, 2025.

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 *ICML*, 2021.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, 2022.

Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. In *NeurIPS*, 2022.

Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training gans. *NeurIPS*, 29, 2016.

Vikash Sehwag, Xianghao Kong, Jingtao Li, Michael Spranger, and Lingjuan Lyu. Stretching each dollar: Diffusion training from scratch on a micro-budget. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 28596–28608, 2025.

Jaeyo Shin, Jiwook Kim, and Hyunjung Shim. Representation alignment for just image transformers is not easier than you think. In *ECCV*, pp. 603–620. Springer, 2026.

Oriane Siméoni, Huy V Vo, Maximilian Seitzer, Federico Baldassarre, Maxime Oquab, Cijo Jose, Vasil Khalidov, Marc Szafraniec, Seungeun Yi, Michaël Ramamonjisoa, et al. Dinov3. *arXiv preprint arXiv:2508.10104*, 2025.

Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

Jaskirat Singh, Xingjian Leng, Zongze Wu, Liang Zheng, Richard Zhang, Eli Shechtman, and Saining Xie. What matters for representation alignment: Global information or spatial structure? *arXiv preprint arXiv:2512.10794*, 2025.

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

Stefan Wager, Sida Wang, and Percy S Liang. Dropout training as adaptive regularization. *NeurIPS*, 26, 2013.

Ziqiao Wang, Wangbo Zhao, Yuhao Zhou, Zekai Li, Zhiyuan Liang, Mingjia Shi, Xuanlei Zhao, Pengfei Zhou, Kaipeng Zhang, Zhangyang Wang, et al. Repa works until it doesn't: Early-stopped, holistic alignment supercharges diffusion training. *arXiv preprint arXiv:2505.16792*, 2025.

Ge Wu, Shen Zhang, Ruijing Shi, Shanghua Gao, Zhenyuan Chen, Lei Wang, Zhaowei Chen, Hongcheng Gao, Yao Tang, Jian Yang, Ming-Ming Cheng, and Xiang Li. Representation entanglement for generation: Training diffusion transformers is much easier than you think. In Danielle Belgrave, Cheng Zhang, Laura N. Montoya, Hsuan-Tien Lin, Razvan Pascanu, Piotr Koniusz, Marzyeh Ghassemi, Nancy Chen, Iván Vladimir Meza Ruíz, and Arturo Loaiza-Bonilla (eds.), *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2025, NeurIPS 2025, San Diego, CA, USA, December 2-7, 2025 / Mexico City, Mexico, November 30 - December 5, 2025*, 2025. URL http://papers.nips.cc/paper_files/paper/2025/hash/0b99315234cc95e6ef281f9155b68832-Abstract-Conference.html.

Shaodong Xu, Zhendong Wang, Litong Gong, Zexian Li, Wengang Zhou, Tiezheng Ge, and Houqiang Li. Beyond point-wise matching: Structural representation alignment for accelerating diffusion transformers. *arXiv preprint arXiv:2605.16949*, 2026.

Sihyun Yu, Sangkyung Kwak, Huiwon Jang, Jongheon Jeong, Jonathan Huang, Jinwoo Shin, and Saining Xie. Representation alignment for generation: Training diffusion transformers is easier than you think. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025. URL https://openreview.net/forum?id=DJSZGGZYVi.

Hongkai Zheng, Weili Nie, Arash Vahdat, and Anima Anandkumar. Fast training of diffusion models with masked transformers. *arXiv preprint arXiv:2306.09305*, 2023.

A Experimental Setup

Training Setup. Table 6 summarizes the default hyperparameter settings of REG + MaskAlign for SiT-B/2 and SiT-XL/2. For REPA + MaskAlign, the class token and its prediction loss are omitted as described in Sec. 5. Following the experimental protocol of REPA (Yu et al., 2025), we train models in the latent space with v-prediction. Unless otherwise specified, we use DINOv2-B (Oquab et al., 2023) as the pretrained vision encoder. For the encoder generalization experiments in Figure 4, we replace DINOv2-B with DINOv3-B (Siméoni et al., 2025) or CLIP-B (Radford et al., 2021) while keeping the other training settings unchanged. The alignment weight is set to $\lambda = 0.5$, and the prediction weight for the class token is set to $\beta = 0.03$. We use AdamW with a batch size of 256 and a learning rate of 1×10^{-4}.

Table 6: Hyperparameter settings across different model scales.

Backbone	SiT-B	SiT-XL
Architecture		
#Params	154M	732M
Input	$32 \times 32 \times 4$	$32 \times 32 \times 4$
Layers	12	28
Hidden dim.	768	1,152
Num. heads	12	16
MaskAlign settings		
β	0.03	0.03
λ	0.5	0.5
Alignment depth	4	8
Mixing Layers	2	2
Mask Ratio	25%	25%
$\text{sim}(\cdot,\cdot)$	cos. sim.	cos. sim.
Encoder $\mathcal{E}_{VF}(I)$	DINOv2-B	DINOv2-B
Optimization		
Batch size	256	256
Optimizer	AdamW	AdamW
lr	0.0001	0.0001
(β_1, β_2)	(0.9, 0.999)	(0.9, 0.999)
Interpolants		
α_t	$1 - t$	$1 - t$
σ_t	t	t
w_t	σ_t	σ_t
Training objective	v-prediction	v-prediction
Sampler	Euler-Maruyama	Euler-Maruyama
Sampling steps	250	250

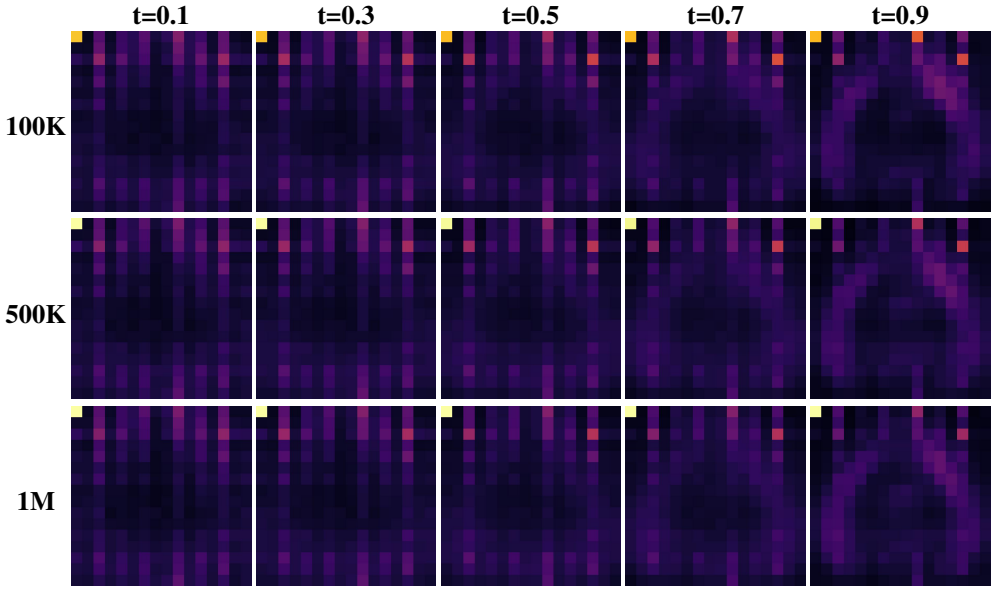
Evaluation Setup. We report FID (Heusel et al., 2017), sFID (Nash et al., 2021), IS (Salimans et al., 2016), precision, and recall (Kynkäänniemi et al., 2019) to evaluate generation quality. All metrics are computed using 50K generated images. Following REPA (Yu et al., 2025), we use the SDE Euler-Maruyama solver with 250 sampling steps. For the evaluation with different feature encoders, we use the registered `dinov2-vit-l-14`, `clip-vit-l-14`, and `vgg16` feature extractors in torch-fidelity (Obukhov et al., 2020).

B Additional Token-Level Alignment Heatmaps

Table 7 shows additional heatmaps of alignment gradients across tokens at different timesteps and training iterations, using DINOv2-B as the vision encoder. Each heatmap shows the probability that each spatial position appears among the top-10% tokens ranked by alignment-gradient norm. Across

different timesteps and checkpoints, tokens with large alignment gradients remain concentrated at particular spatial positions, further supporting our observation that full-token representation alignment does not affect all patch tokens uniformly.

Table 7: Additional alignment-gradient heatmaps across timesteps and training iterations. Rows denote training iterations, and columns denote timesteps. Each heatmap shows the spatial probability of top-10% alignment-gradient tokens, using the same color range $[0, 0.8]$.



C Additional Ablation Studies

Effect of Mask Ratio. We train models with different mask ratios for 400K iterations. As shown in Table 8, a moderate mask ratio of 0.25 achieves the best performance, reducing FID from 3.52 without masking to 2.84. Increasing the mask ratio to 0.5 raises FID to 3.15, while an excessively high mask ratio of 0.75 severely degrades performance. These results suggest that random token masking should provide sufficient perturbations to the token set to regularize alignment, while still preserving enough input information for stable training.

Effect of Mixing Layers. We study the effect of the number of token mixing layers before masking. As shown in Table 9, using two mixing layers achieves the best performance, reducing FID to 2.84. With only one mixing layer, the model obtains a higher FID of 3.23, suggesting that one mixing layer does not sufficiently compensate for the information lost through masking. Using three mixing layers also degrades performance, with FID increasing to 3.02, indicating that additional mixing does not provide further benefit. Two token mixing layers perform best among the tested settings.

D Additional Results on ImageNet

Table 10 reports additional quantitative results of MaskAlign on ImageNet 256×256 without classifier-free guidance (CFG). We evaluate MaskAlign at different training iterations to provide a more detailed view of its convergence behavior. As training proceeds, MaskAlign consistently improves generation quality, reducing FID from 22.36 at 50K iterations to 2.38 at 1M iterations. Compared with REG trained for 1M iterations, MaskAlign achieves better FID, sFID, and IS at the same training budget, while maintaining comparable precision and recall.

Table 8: Ablation study on the mask ratio. All models are trained for 400K iterations without CFG.

Mask Ratio	FID↓	sFID↓	IS↑
0	3.52	4.90	184.13
0.25 (Ours)	**2.84**	**4.85**	**194.57**
0.5	3.15	5.08	188.38
0.75	5.82	5.29	152.28

Table 9: Ablation study on the number of pre-mask token mixing layers. All models are trained for 400K iterations without CFG.

Mixing Layers	FID↓	sFID↓	IS↑
1	3.23	4.93	188.49
2 (Ours)	**2.84**	**4.85**	**194.57**
3	3.02	4.88	192.54

Table 10: Additional quantitative results of REG + MaskAlign on ImageNet 256×256 without classifier-free guidance (CFG). We report FID, sFID, IS, precision, and recall across different training iterations.

Model	#Params	Iter.	FID↓	sFID↓	IS↑	Prec.↑	Rec.↑
SiT-XL/2	675M	7M	8.3	6.32	131.7	0.68	**0.67**
REG	677M	1M	2.7	4.93	201.8	0.76	0.66
REG + MaskAlign	732M	50K	22.36	20.62	70.24	0.63	0.53
REG + MaskAlign	732M	110K	6.34	6.47	145.79	0.75	0.58
REG + MaskAlign	732M	200K	3.98	5.18	172.60	0.77	0.60
REG + MaskAlign	732M	400K	2.84	4.85	194.57	0.77	0.62
REG + MaskAlign	732M	600K	2.67	4.78	198.10	**0.77**	0.64
REG + MaskAlign	732M	1M	**2.38**	**4.78**	**205.37**	0.76	0.65

E ADDITIONAL QUALITATIVE RESULTS

We present additional qualitative results of SiT-XL/2 + REG + MaskAlign in Figures 5–9.



Figure 5: Generated samples. The class label is “great grey owl” (24).



Figure 6: Generated samples. The class label is “golden retriever” (207).



Figure 7: Generated samples. The class label is “arctic wolf” (270).



Figure 8: Generated samples. The class label is “polecat” (358).



Figure 9: Generated samples. The class label is “castle” (483).