Flowing Backwards: Improving Normalizing Flows via Reverse Representation Alignment

Yang Chen[1,2], Xiaowei Xu[2], Shuai Wang[1], Chenhui Zhu[1,2], Ruxue Wen[2], Xubin Li[2], Tiezheng Ge[2], Limin Wang[1,3,✉]

[1]State Key Laboratory for Novel Software Technology, Nanjing University
[2]Alibaba Group
[3]Shanghai AI Lab

Abstract

Normalizing Flows (NFs) are a class of generative models distinguished by a mathematically invertible architecture, where the forward pass transforms data into a latent space for density estimation, and the reverse pass generates new samples from this space. This characteristic creates an intrinsic synergy between representation learning and data generation. However, the generative quality of standard NFs is limited by poor semantic representations from log-likelihood optimization. To remedy this, we propose a novel alignment strategy that creatively leverages the invertibility of NFs: instead of regularizing the forward pass, we align the intermediate features of the generative (reverse) pass with representations from a powerful vision foundation model, demonstrating superior effectiveness over naive alignment. We also introduce a novel training-free, test-time optimization algorithm for classification, which provides a more intrinsic evaluation of the NF's embedded semantic knowledge. Comprehensive experiments demonstrate that our approach accelerates the training of NFs by over 3.3×, while simultaneously delivering significant improvements in both generative quality and classification accuracy. New state-of-the-art results for NFs are established on ImageNet 64×64 and 256×256.

1 Introduction

Normalizing Flows (NFs) represent a distinct class of generative models, characterized by their exact mathematical invertibility (Rezende and Mohamed 2015; Dinh, Krueger, and Bengio 2014; Dinh, Sohl-Dickstein, and Bengio 2016). This property defines their rigid, dual-pathway architecture: a forward pass transforms data into a simple latent distribution for exact log-likelihood optimization, while a mathematically precise reverse pass generates data from that same latent space (Figure 1a,b). This structure implies the synergy of NFs between data generation and representation learning, where the two are truly two sides of the same coin.

This inherent synergy suggests a clear path toward enhancing the generative capabilities of NFs: by improving the quality of their learned representations. However, this potential remains largely underexploited. Standard NFs, optimized solely for log-likelihood on the forward pass, often fail to learn semantically meaningful features, which in turn limits their generative quality. The model's rigid adherence to the likelihood objective prevents it from fully realizing the benefits of its own architectural duality.

This line of inquiry is particularly relevant given recent findings that actively improving a model's representational quality can enhance its generative capabilities. For instance, a notable method, REPA (Yu et al. 2024), regularized the internal features of diffusion models against a strong, pretrained visual encoder. This 'representation-first' strategy yielded significant gains in both training efficiency and generation quality, demonstrating the effectiveness of leveraging high-quality external guidance.

Inspired by REPA, our work aims to adapt and extend it to the unique context of NFs. We ask: **how can the invertible structure of NFs be leveraged to effectively synergize representation learning and generation?** By utilizing the reverse generative pathway for alignment—a strategy uniquely enabled by NFs—we move beyond direct application to explore new alignment strategies within this model class. This forms the core motivation for our work.

In this paper, we first need a way to probe the intrinsic discriminative abilities of a given NF. Departing from the standard linear-probe protocol, we introduce a novel **training-free, test-time optimization** algorithm, visually depicted in Figure 1c. Instead of training an auxiliary classifier, our method directly leverages the model's own loss landscape to perform classification, providing a more direct measure of its embedded semantic knowledge. Our initial evaluation using this framework confirms a critical weakness: standard NFs, despite their generative prowess, exhibit poor discriminative performance.

To address this deficiency, we propose reverse representation alignment, a novel strategy built upon two core components: representation alignment and the exploitation of the model's reverse pass. We term the act of operating on the generative pathway *'flowing backwards'*. In this process, our method directly enforces semantic consistency by aligning the intermediate features during the true generative pass ($\mathbf{z}$-to-$\mathbf{x}$), a concept uniquely enabled by the invertible architecture of NFs.

As conceptually illustrated in Figure 1d, our comprehen-

✉ : Corresponding author (lmwang@nju.edu.cn).

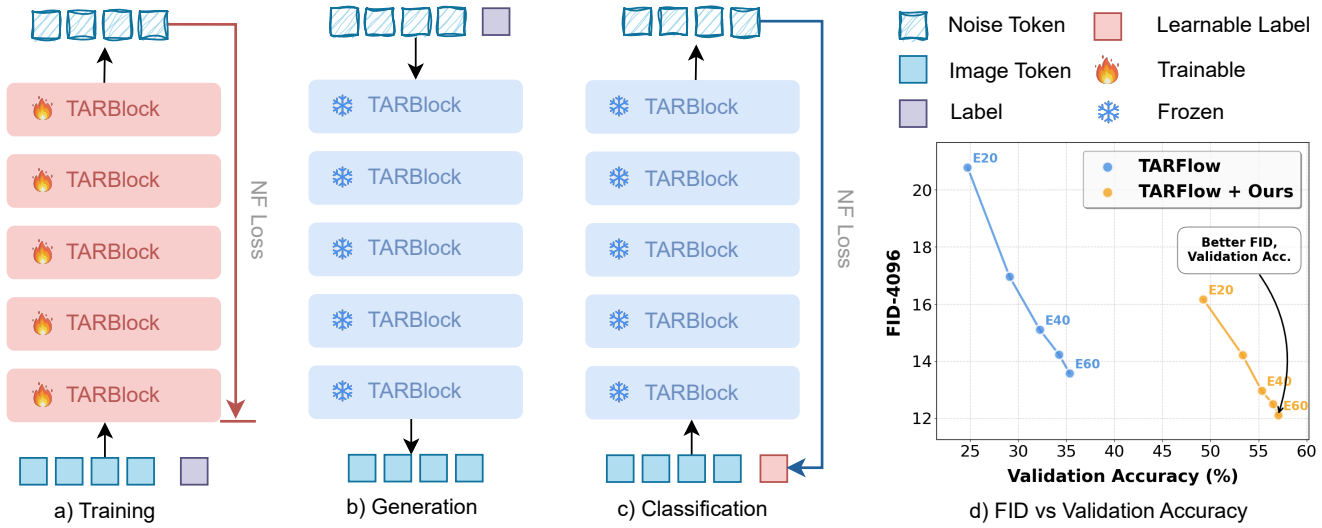

Figure 1: Take TARFlow as a representative NF. (a) Training process maps images to a noise distribution. (b) The reverse pass generates images. (c) Optimizing a label token by NF loss to classify. (d) The FID-Accuracy plot demonstrates that our representation alignment improves both generation quality and classification performance.

sive experiments show that this reverse alignment strategy is remarkably effective. It not only yields substantial improvements in generative quality as measured by FID but also, when assessed with our test-time method, unlocks a dramatic increase in classification accuracy. In summary, our main contributions are:

- We design and systematically evaluate several alignment strategies for NFs, culminating in our proposed **reverse representation alignment** (R-REPA) method that uniquely leverages the model's invertibility.
- We propose a novel training-free, test-time optimization method for NF-based classification (Figure 1c), serving as both a diagnostic tool and a more intrinsic evaluation metric.
- We demonstrate empirically that our approach significantly enhances both the generative and discriminative performance of NFs (Figure 1d), establishing a new state-of-the-art for synergizing these two capabilities.

2 Related Work

Normalizing Flows. NFs are exact likelihood-based generative models (Dinh, Krueger, and Bengio 2014; Rezende and Mohamed 2015). While historically surpassed by diffusion models (Ma et al. 2024; Wang et al. 2024, 2025b,a) in sample quality, recent work has revitalized the field. TARFlow (Zhai et al. 2024) introduces a Transformer-based architecture that achieves state-of-the-art likelihoods and generates samples with quality comparable to diffusion models. JetFormer (Tschannen, Pinto, and Kolesnikov 2024) leverages an NF as a core, jointly trained component within a unified autoregressive model for high-fidelity joint image-text generation, eliminating the need for pre-trained autoencoders. Our concurrent work STARFlow (Gu et al. 2025) successfully scales NFs in terms of both model capacity and task complexity. These works demonstrate the renewed potential of NFs when integrated with modern architectures.

Representation Alignment for Generation. A recent paradigm for accelerating generative model training is Representation Alignment, which leverages features from pre-trained vision foundation models (VFMs) as guidance. The seminal **REPA** (Yu et al. 2024) introduced a loss to align the internal hidden states of a denoising network with VFM features, drastically improving convergence and final sample quality. This simple yet powerful principle has proven highly effective and was quickly extended. Subsequent work has used it to enable stable end-to-end training of the entire latent diffusion pipeline (Lee, Park, and Kim 2024), to improve the VAE's latent space directly (Zheng et al. 2024), and to adapt the alignment strategy to other backbones like U-Nets (Tian et al. 2024b), demonstrating its broad utility.

3 Preliminary: Normalizing Flow and TARFlow

Normalizing Flows (NFs) (Rezende and Mohamed 2015; Dinh, Krueger, and Bengio 2014; Dinh, Sohl-Dickstein, and Bengio 2016) represent a class of generative models based on exact likelihood built on the formula for change of variables. Given a continuous data distribution p_{data} for inputs $\mathbf{x} \in \mathbb{R}^D$, a Normalizing Flow learns a parametric, invertible transformation $f_\theta : \mathbb{R}^D \to \mathbb{R}^D$. This function maps complex data $\mathbf{x}$ to a simple, tractable base distribution p_0 (e.g., a standard Gaussian), often referred to as the noise or latent space. The model is trained by maximizing the log-likelihood (MLE) of the data:

$$\max_\theta \mathbb{E}_{\mathbf{x}\sim p_{\text{data}}}\left[\log p_0(f_\theta(\mathbf{x})) + \log\left|\det \frac{\partial f_\theta(\mathbf{x})}{\partial \mathbf{x}}\right|\right], \quad (1)$$

where the first term encourages mapping data points to high-density regions of the prior p_0, while the log-determinant of

Figure 2: Selected Samples on ImageNet 256 $\times$ 256 from L-TARFlow + R-REPA. We use classifier-free guidance equal to 2.0.

the Jacobian term penalizes excessive local volume shrinkage, ensuring the transformation remains bijective. A generative model is obtained by the inverse transformation f_θ^{-1}, with a sampling procedure $\mathbf{x} = f_\theta^{-1}(\mathbf{z})$, where $\mathbf{z} \sim p_0(\mathbf{z})$.

A prominent and computationally efficient variant is the Autoregressive Flow (AF) (Kingma et al. 2016; Papamakarios, Murray, and Pavlakou 2017). In an AF, the transformation f_θ is directly defined by a pair of parameter-generating functions $(\mu_\theta, \sigma_\theta)$, which specify an element-wise affine map. Crucially, these functions are architecturally constrained to be autoregressive: the parameters for each dimension d are computed using only the preceding input dimensions $\mathbf{x}_{<d}$. The forward (encoding) and inverse (sampling) processes for each dimension $d \in [1, D]$ are:

$$\begin{aligned} \mathbf{z}_d &= (\mathbf{x}_d - \mu_\theta(\mathbf{x}_{<d})) \odot \sigma_\theta(\mathbf{x}_{<d})^{-1}, \\ \mathbf{x}_d &= \mu_\theta(\mathbf{x}_{<d}) + \sigma_\theta(\mathbf{x}_{<d}) \odot \mathbf{z}_d. \end{aligned} \tag{2}$$

This autoregressive structure ensures the Jacobian of f_θ is triangular, greatly simplifying the log-determinant term in Eq. 1 to a simple sum: $-\sum_{d=1}^{D} \log \sigma_\theta(\mathbf{x}_{<d})$.

Recently, TARFlow (Zhai et al. 2024) was introduced as a high-performance NF architecture for image data, building upon the AF framework. TARFlow is constructed by stacking multiple Transformer AutoRegressive Blocks (TARBlocks), $\mathbf{z} = f_\theta^T \circ \cdots \circ f_\theta^1(\mathbf{x})$, where each block f_θ^t processes its input using a different autoregressive ordering π^t. By alternating these orderings, the stacked blocks can capture dependencies across all dimensions. The parameters μ and σ for each block are modeled using causal Transformer layers. Assuming the final output $\mathbf{z} = \mathbf{x}^T$ follows a standard Gaussian prior, the end-to-end training objective becomes:

$$\max_\theta \mathbb{E}_{\mathbf{x}\sim p_{\text{data}}}\Big[-\frac{1}{2}\|\mathbf{z}\|_2^2 - \sum_{t=1}^{T}\sum_{d=1}^{D} \log \sigma_\theta^t(\mathbf{x}^{t-1}_{\pi^t_{<d}}) \Big], \tag{3}$$

where $\mathbf{x}^t = f_\theta^t(\mathbf{x}^{t-1})$ and $\mathbf{x}^0 = \mathbf{x}$. Additionally, TARFlow employs noise augmented training and score-base denoising to improve the modeling capability.

4 Methodology

In this section, we present our proposed methods for enhancing and leveraging Normalizing Flows. We begin by introducing a novel, training-free classification algorithm that directly utilizes the learned probability density from class-conditional NFs. Subsequently, we describe a representation alignment algorithm specifically designed to improve the discriminative quality of the latent representations within NFs. Finally, to address the computational challenges of modeling high-dimensional data, we adapt the TARFlow architecture to operate within a compressed latent space, enabling efficient generation.

Algorithm 1: Training-Free Classification via Single-Step Gradient.

Require: Image $\mathbf{x}$, pre-trained model f_θ, class embeddings $\mathbf{E} \in \mathbb{R}^{K \times D_{\text{emb}}}$.
Ensure: Predicted class label y_{pred}.

Initialize logits $\boldsymbol{\lambda} \leftarrow \mathbf{0} \in \mathbb{R}^K$.
Compute weighted class embedding:
$\mathbf{p} \leftarrow \text{softmax}(\boldsymbol{\lambda})$
$\mathbf{e}_{\text{eff}} \leftarrow \mathbf{p}^T\mathbf{E}$
Compute the log-likelihood score:
$\mathcal{L}(\boldsymbol{\lambda}) \leftarrow \log p(\mathbf{x} \mid \mathbf{e}_{\text{eff}}; \theta)$
Compute the gradient with respect to the logits:
$\mathbf{g} \leftarrow \nabla_{\boldsymbol{\lambda}}\mathcal{L}(\boldsymbol{\lambda})$
Predict the class corresponding to the largest gradient component:
$y_{\text{pred}} \leftarrow \text{argmax}_k(\mathbf{g})_k$
return y_{pred}

4.1 Training-Free Classification with NFs

We introduce a novel, training-free classification algorithm that leverages the density estimation capabilities of a pre-trained class-conditional TARFlow model. Instead of training a separate classifier, our method reframes classification as an inference-time optimization problem. The core idea is to find the class label y that maximizes the conditional log-likelihood $\log p(\mathbf{x}|y;\theta)$ for a given input image $\mathbf{x}$, where θ are the frozen parameters of the generative model.

We achieve this by estimating the gradient of the log-likelihood with respect to a"soft" class conditioning. This is

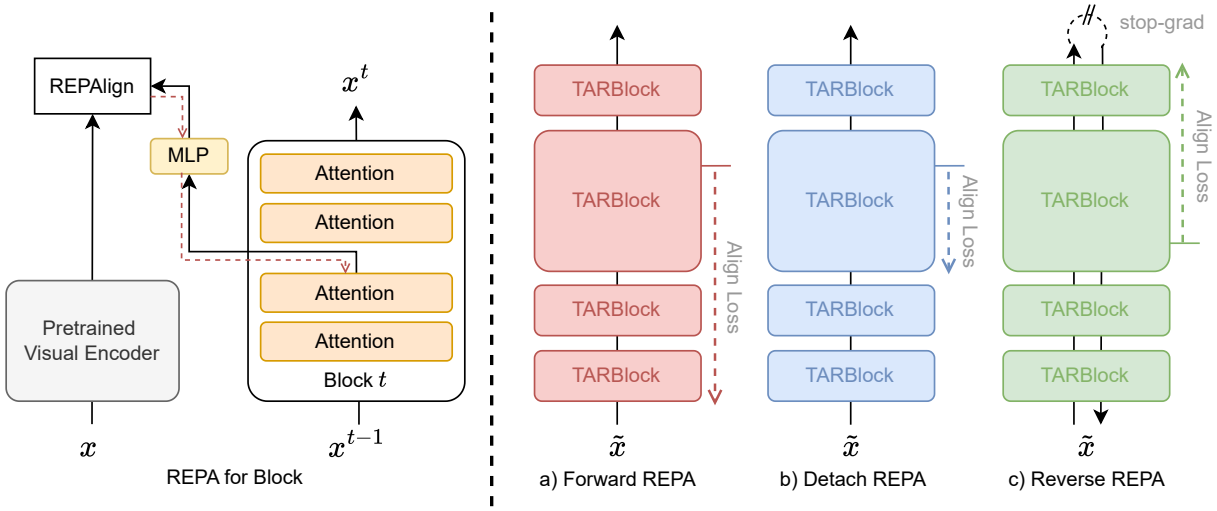

Figure 3: An overview of our Representation Alignment (REPA) mechanism. **Left:** Intermediate features from a TARFlow block are projected by an MLP and aligned with features from a pre-trained visual encoder. **Right:** The three gradient backpropagation strategies explored: (a) Forward REPA (F-REPA), updating all preceding blocks; (b) Detach REPA (D-REPA), updating only the current block; and (c) Reverse REPA (R-REPA), which leverages the *inverse (generative) computational graph* to update all subsequent blocks. While we depict alignment at a single location for clarity, this mechanism can be applied concurrently across multiple layers.

done by first defining a set of classification logits, $\boldsymbol{\lambda} \in \mathbb{R}^K$, which control a weighted average of the model's class embeddings $\mathbf{E}$. The gradient of the log-likelihood is then computed with respect to these logits at their initialization point. The class with the largest gradient component is selected as the prediction, as it indicates the direction of greatest increase in likelihood. The entire process requires only a single forward and backward pass through the model and is detailed in Algorithm 1.

4.2 Representation Alignment

While the MLE objective excels at density modeling, the learned intermediate features of a NF are not inherently optimized to be semantically meaningful. To address this, we introduce a feature alignment mechanism that injects high-level semantic guidance from a powerful, pre-trained vision model into the TARFlow's generative process.

As illustrated in Figure 3 (left), we use a pre-trained, frozen vision encoder $\Phi(\cdot)$ to extract a semantic representation $\mathbf{v} = \Phi(\mathbf{x}) \in \mathbb{R}^{P \times D}$ from an input image $\mathbf{x}$, where P is the number of patches and D is the embedding dimension. The objective is to align TARFlow's intermediate features with this target representation $\mathbf{y}$.

Let $\mathbf{h}^{(t,l)}$ be the feature map from layer l of block t within TARFlow. We project these features into the semantic space using a learnable head, Proj_ϕ (a simple MLP). The alignment loss then maximizes the patch-wise similarity between the projected and target features:

$$\mathcal{L}_{\text{align}}^{(t,l)}(\theta, \phi) := -\frac{1}{P} \sum_{p=1}^{P} \text{sim}\left(\mathbf{v}^{[p]}, \left[\text{Proj}_\phi\left(\mathbf{h}^{(t,l)}\right)\right]^{[p]}\right), \tag{4}$$

where p is the patch index and $\text{sim}(\cdot,\cdot)$ is a similarity function, such as cosine similarity. This alignment can be flexibly applied to any set of layers $\mathcal{A} = \{(t_1, l_1), \dots\}$.

Crucially, we explore three distinct strategies for backpropagating the gradient of this alignment loss, each manipulating the computational graph to control how the parameters are updated by the alignment loss. These strategies are visualized in Figure 3 (right).

Forward Strategy. As the most direct approach, this strategy involves backpropagating the gradient of $\mathcal{L}_{\text{align}}^{(t,l)}$ through the forward computational graph. As illustrated in Figure 3 (a), this updates both the projector ϕ and all parameters of the TARFlow layers preceding layer (t, l).

Detach Strategy. This strategy draws an analogy to diffusion models, treating each TARFlow block as a network operating at a specific timestep t. To isolate the alignment process to this single 'timestep', we detach the input to the block. Consequently, the gradient only updates the parameters within that block (i.e., θ_t) and the projector ϕ, preventing any influence on preceding blocks (Figure 3, b).

Reverse Strategy. This novel strategy fundamentally alters the update mechanism by leveraging the computational graph of the *reverse (generative) process*. Specifically, we first compute the latent variable $\mathbf{z} = f_\theta(\mathbf{x})$ via the forward pass and then detach it. A new computational graph is then constructed by executing the inverse flow f_θ^{-1}, starting from the detached latent $\mathbf{z}_{\text{detached}}$. The alignment loss is computed within this inverse pass. Crucially, backpropagation from this loss occurs entirely on the generative graph, inherently confining gradient updates to the parameters of layers subsequent to the alignment layer (t, l) (relative to the original

forward pass). Figure 3 (c) conceptualizes this, showing how the stop-gradient on $\mathbf{z}$ reroutes the gradient path exclusively through the generative pathway.

Final Loss Formulation. The total training objective is a weighted sum of the NF loss and the averaged alignment losses from all chosen layers in the set $\mathcal{A}$:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{NF}} + \lambda_{\text{align}} \left(\frac{1}{|\mathcal{A}|} \sum_{(t,l)\in\mathcal{A}} \mathcal{L}_{\text{align}}^{(t,l)}(\theta, \phi) \right), \quad (5)$$

where λ_{align} is a hyperparameter balancing the two terms. The gradient computation for $\mathcal{L}_{\text{align}}$ follows one of the three strategies outlined above.

4.3 Moving to Latent Space

To scale our method to high-resolution synthesis, we transition from modeling pixels directly to modeling the latent space of a pre-trained Variational Autoencoder (VAE). This established strategy allows us to offload the task of low-level perceptual compression. Our primary work thus focuses on the core challenge: applying the powerful density estimation and refinement techniques of TARFlow to the compact and semantically-rich latent codes.

The training process is adapted to this latent space. For a clean latent vector x obtained from the VAE encoder, our NF model, f_θ, is trained on a noisy version $\tilde{x}$:

$$\tilde{x} = x + \epsilon, \quad \text{where } \epsilon \sim \mathcal{N}(0, \sigma^2 I). \quad (6)$$

The model thus learns to estimate the density $p_\theta(\tilde{x})$.

The generative process mirrors this by first sampling from this learned noisy distribution and then applying a denoising step. Specifically, a noisy latent sample $\tilde{x}$ is generated via the inverse transformation, $\tilde{x} = f_\theta^{-1}(z)$, where $z \sim \mathcal{N}(0, I)$ is a sample from the base distribution. This sample is subsequently refined using the score-based denoising procedure:

$$\hat{x} = \tilde{x} + \sigma^2 \nabla_{\tilde{x}} \log p_\theta(\tilde{x}). \quad (7)$$

Finally, the refined latent vector $\hat{x}$ is decoded into the final image, effectively scaling the precise likelihood modeling and powerful sample refinement of TARFlow to the high-resolution domain.

5 Experiments

Dataset and Task. We conduct our class-conditional image generation and classification experiments on the ImageNet-1K dataset (Deng et al. 2009). Our models are trained exclusively on the training set and evaluated at two distinct resolutions: 64×64 and 256×256.

Evaluation Metrics. To assess the performance of our generative model, we employ a standard suite of metrics to measure sample fidelity and diversity: Fréchet inception distance (FID) (Heusel et al. 2017), sFID (Nash et al. 2021), and Inception Score (IS) (Salimans et al. 2016). Following TARFlow, we sample 4096 images for evaluation. Besides, we report results at the optimal CFG scale for each model, determined via a grid search detailed in Figure 4. To evaluate the discriminative ability, we measure the classification accuracy on the ImageNet-1K validation set. This is achieved using our proposed test-time optimization classification.

Type	Blocks	Layers	FID↓	sFID↓	IS↑	Acc.(%)↑
TARFlow (Zhai et al. 2024)			12.91	33.79	36.62	37.43
*Setup Group 1: Alignment applied to **All** blocks*						
Forward	All	2	12.25	37.97	40.85	46.97
Detach	All	2	12.19	34.31	41.98	49.06
Reverse	All	2	12.21	33.80	42.08	49.91
*Setup Group 2: Alignment applied to selected **2** blocks*						
Forward	1 & 2	2	12.67	39.99	41.11	61.16
Detach	1 & 2	2	12.73	33.84	37.81	**61.63**
Detach	7 & 8	2	12.12	34.00	41.18	55.14
Reverse	7 & 8	2	11.93	33.78	40.90	55.21
Setup Group 3: Ablation of alignment layers						
Reverse	7 & 8	2	11.93	33.78	40.90	55.21
Reverse	7 & 8	4	11.84	**33.61**	**46.06**	58.91
Reverse	7 & 8	6	**11.71**	33.68	44.31	57.35

Table 1: Ablation study of our proposed alignment method on ImageNet 64×64, evaluated at 400k training iterations. We vary the alignment *Type* and which blocks and layers to apply it to. Best results for each metric are in **bold**.

Implementation Details. For experiments at the 64×64 resolution, our model architecture strictly adheres to the design of TARFlow (Zhai et al. 2024). Specifically, the model is composed of 8 TARBlocks. Each block, in turn, contains 8 layers of causal attention. The channel dimension is set to 1024, and the model operates on non-overlapping image patches of size 4×4.

For the higher resolution of 256×256, we first leverage a pre-trained VAE-ft-EMA (Esser, Rombach, and Ommer 2021) to compress images into a lower-dimensional latent space. Our generative model then operates on this latent representation. The transformer architecture is enhanced with two key components: Rotary Position Embeddings (RoPE) (Su et al. 2024) and the SwiGLU activation function (Touvron et al. 2023a,b). To maintain consistent patch number with 64×64, we use a patch size of 2×2. All other hyperparameters, including the number of TARBlocks (8), layers per block (8), and channel dimension (1024), are kept consistent with the 64×64 configuration.

5.1 Ablation Studies

In this section, we conduct a series of ablation studies to systematically determine our optimal model configuration. We first investigate the core design of the REPA mechanism (its backpropagation strategy, location, and depth), and then analyze key hyperparameters for training and sampling. Unless otherwise specified, all ablations are performed on ImageNet 64×64 at 400k training iterations.

Backpropagation Strategy. As shown in Table 1, the choice of backpropagation strategy proves to be critical. We find the Forward strategy consistently degrades sFID (e.g., from 33.79 to 37.97). We hypothesize that this is because its unconstrained gradient flow creates a tension between the

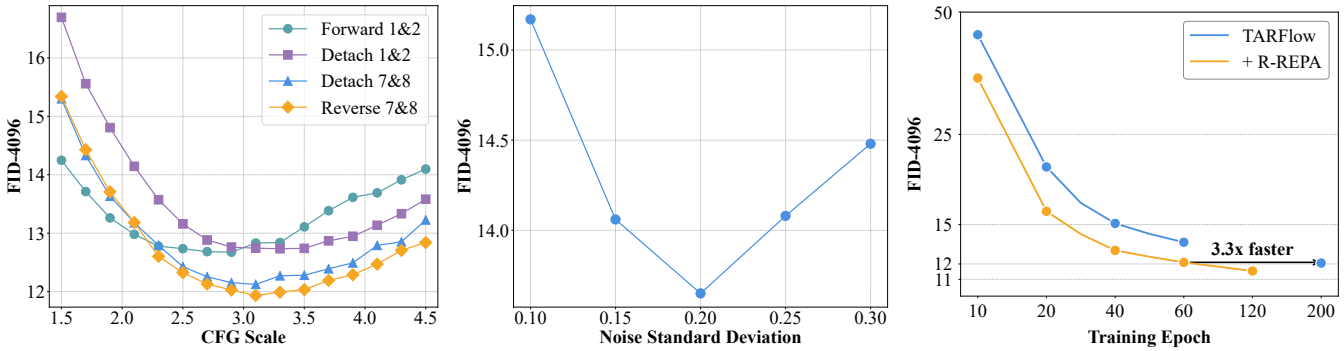

Figure 4: Hyperparameters Ablations and Training Convergence. **Left:** CFG search results on ImageNet 64×64. The Reverse REPA strategy applied to later model blocks yields the best performance across various CFG scales. **Center:** Ablation of noise standard deviation on latent space. We identify an optimal noise standard deviation of 0.20. **Right:** Reverse REPA improves sample fidelity and accelerates training convergence by 3.3× on ImageNet 64×64.

MLE objective and the alignment loss in the model's early blocks. Forcing these foundational blocks, which are likely specialized for low-level spatial statistics, to also conform to high-level semantics proves detrimental to overall sample coherence. The 'Detach' strategy improves upon this by localizing updates. However, the 'Reverse' strategy is demonstrably superior, achieving a better FID (11.93 vs. 12.12) in direct comparison. By updating only the generative path (f_θ^{-1}), it effectively refines synthesis without disrupting the core density model.

Alignment Block and Layer. The choice of which blocks to align creates a trade-off between generative quality and feature semantics. Aligning Blocks 7 & 8, which are the first to operate on the latent variable $\mathbf{z}$ during the generative process, yields the best FID (11.93). Guiding these initial synthesis steps helps establish a strong high-level image structure. Conversely, aligning Blocks 1 & 2 produces the best semantic accuracy (**61.63%**). These blocks are the first to process the input image $\mathbf{x}$ during encoding, so aligning them directly optimizes feature extraction but harms FID by constraining the delicate final synthesis steps.

Finally, within our best-performing setup (R-REPA, aligning 7 & 8 blocks), we found that guiding deeper layers leads to better fidelity. As shown in Group 3, the FID progressively improves from 11.93 to **11.71** as we move alignment from the 2nd to the 6th layer. This suggests that guiding more refined features closer to a block's output provides a more potent refinement signal. While an intermediate (4th) layer achieves the best IS (**46.06**), the deepest (6th) layer is the clear choice for maximizing the final image quality.

Classifier-Free Guidance (CFG) Scale. We optimized the CFG scale, a critical hyperparameter, for each model to ensure a fair comparison. As shown in Figure 4, our final configuration ('Reverse 7&8') achieves the lowest FID while also demonstrating robustness across a wide scale range, with an optimum near 3.1. All results are reported at their respective optimal scales.

Model	Res.	Iter.	FID-4096↓	Acc.(%)↑
TARFlow	64	1M	11.76	39.97
+R-REPA (Ours)	64	**400K**	**11.71**	**57.76**
+R-REPA (Ours)	64	**600K**	**11.53**	**57.75**
+R-REPA (Ours)	64	**800K**	**11.48**	**57.65**
+R-REPA (Ours)	64	**1M**	**11.25**	**57.02**
Latent-TARFlow	256	400K	13.82	45.97
Latent-TARFlow	256	1M	13.05	40.22
+R-REPA (Ours)	256	**400K**	**13.26**	**57.85**
+R-REPA (Ours)	256	**1M**	**12.79**	**56.24**

Table 2: Training progress comparison. Our method, **+R-REPA**, consistently outperforms the vanilla TARFlow baselines across different checkpoints on both ImageNet 64×64 and 256×256 resolutions.

Latent Noise Std. For the 256×256 latent-space model, the standard deviation (σ) of noise added to VAE latents (Eq. 6) is crucial. We ablated this value, with results in Figure 4. A clear performance peak exists at $\sigma = 0.20$, which we use for all latent-space experiments.

5.2 Main Results

Equipped with the optimal R-REPA configuration from our ablations, we now evaluate its performance against baselines and state-of-the-art models on ImageNet at 64×64 and 256×256 resolutions.

Performance against Baselines. We first compare our method, R-REPA, directly against the vanilla TARFlow baselines. As shown in Table 2, our method provides substantial improvements in both sample quality (FID) and learned feature discriminability (Accuracy).

On ImageNet 64×64, our final model achieves an FID of **11.25** and an accuracy of **57.02%**, significantly outperforming the baseline's 11.76 FID and 39.97% accuracy. The faster convergence of discriminative accuracy over gener-

Model	FID↓	sFID↓
Diffusion Models / Flow Matching		
EDM (Karras et al. 2022)	1.36	–
iDDPM (Nichol and Dhariwal 2021)	2.92	3.79
ADM (Dhariwal and Nichol 2021)	2.09	4.29
Generative Adversarial Networks (GANs)		
IC-GAN (Casanova et al. 2021)	6.70	–
BigGAN (Brock, Donahue, and Simonyan 2018)	4.06	3.96
Consistency Models (CMs)		
CD (LPIPS) (Song et al. 2023)	4.70	–
iCT-deep (Song and Dhariwal 2023)	3.25	–
Normalizing Flows		
TARFlow (Zhai et al. 2024)†	4.21	5.34
+ R-REPA (Ours)	**3.69**	**4.34**

Table 3: Image generation results on class-conditional ImageNet 64×64. We report FID and sFID with 50K samples. †Result obtained using their officially released codebase.

Model	FID↓	sFID↓	IS↑
Diffusion Models			
ADM (Dhariwal and Nichol 2021)	4.59	5.25	186.70
CDM (Ho et al. 2022)	4.88	–	158.71
LDM-4 (Rombach et al. 2022)	3.60	7.51	247.67
DiT (Peebles and Xie 2023)	2.27	4.60	278.24
SiT (Ma et al. 2024)	2.06	4.50	270.30
Autoregressive (discrete)			
RQ-Trans. (Lee et al. 2022)	3.80	–	323.7
LlamaGen-3B (Sun et al. 2024)	2.18	–	263.33
VAR (Tian et al. 2024a)	1.73	–	350.2
Autoregressive (continuous)			
MAR-AR (Li et al. 2024)	4.69	–	244.6
MAR (Li et al. 2024)	1.55	–	303.7
DART (Gu et al. 2024)	3.82	–	263.8
Normalizing Flow			
Latent-TARFlow	5.15	6.78	243.49
+R-REPA (Ours)	**4.95**	**6.89**	**234.99**
+Patch Size 1 (Ours)	**4.18**	**4.96**	**240.8**

Table 4: Class-conditional generation on ImageNet 256×256. We report FID, sFID, and IS with 50K samples. Lower is better for ↓, higher is better for ↑.

ative quality indicates that the model learns high-level semantics early in training before progressively refining fine-grained details for synthesis. This efficiency is a crucial advantage of our approach and leads to accelerated training ($\mathbf{3.3\times}$), as illustrated in Figure 4. Quantitatively, our model at just **400k iterations** already surpasses the fully-trained (1M iter.) baseline in both FID (11.71 vs. 11.76) and, most notably, accuracy (57.76% vs. 39.97%).

A similar leap is observed on the 256×256 latent-space task, where FID improves from 13.05 to **12.79** and accuracy jumps from 40.22% to **56.24%**. This demonstrates that representation alignment not only enhances final performance but also provides a more efficient training signal, leading to superior models in significantly less time.

Generation on ImageNet 64×64. As shown in Table 3, our R-REPA strategy delivers state-of-the-art generative performance for Normalizing Flows on the class-conditional ImageNet 64×64 benchmark. Our method significantly improves upon the strong TARFlow baseline, reducing the FID from 4.21 to **3.69** and the sFID from 5.34 to **4.34**. This performance not only surpasses established GANs like BigGAN (FID 4.06) but also brings flow-based models into closer competition with powerful diffusion models such as iDDPM (FID 2.92). Crucially, this top-tier result is achieved with just **two sampling steps**, highlighting the exceptional inference efficiency of our method compared to the multi-step iterative process required by competing paradigms.

Generation on ImageNet 256×256. We further test the scalability of our approach on the challenging ImageNet 256×256 benchmark by operating in the latent space of a pre-trained VAE. As presented in Table 4, our method again demonstrates remarkable effectiveness. Our optimized configuration, which combines R-REPA with an architectural adjustment to a 1x1 patch size, achieves a highly competitive FID of **4.18** and sFID of **4.96**—a substantial improvement over the baseline. Even the direct application of R-REPA provides a clear boost, reducing FID to 4.95.

Most importantly, these competitive high-resolution results are achieved while **preserving the two-step sampling efficiency**. This demonstrates that the benefits of our approach are not confined to smaller scales but are robust and scalable. By delivering high-fidelity results with a minimal computational budget, our work establishes REPA-enhanced Normalizing Flows as a compelling and highly efficient paradigm for high-resolution image synthesis.

6 Conclusion

In this work, we introduce R-REPA, a novel training strategy that enhances the semantic awareness of NFs. It leverages their unique invertibility to enforce semantic consistency directly on the generative ($\mathbf{z}$-to-$\mathbf{x}$) pass, thereby unlocking the powerful synergy between representation learning and generation inherent in the architecture. The empirical results are compelling. R-REPA establishes a new state-of-the-art for NFs on ImageNet by delivering simultaneous gains in generative fidelity (FID) and classification accuracy over the strong TARFlow baseline. This accuracy gain is rigorously quantified by our novel training-free classification algorithm—a more intrinsic probe of the model's learned semantics. Furthermore, our method demonstrates robust high-resolution scalability while also dramatically boosting training efficiency by over $3.3\times$. Ultimately, our work establishes a powerful new principle for advancing NFs: that fostering a virtuous cycle between semantic representation and the generative process is a direct and effective route to higher fidelity.

Acknowledgements

This work is supported by the National Key R&D Program of China (No. 2022ZD0160900), the Natural Science Foundation of Jiangsu Province (No. BK20250009), the Collaborative Innovation Center of Novel Software Technology and Industrialization, Alibaba Group through Alibaba Innovative Research Program.

References

Brock, A.; Donahue, J.; and Simonyan, K. 2018. Large scale GAN training for high fidelity natural image synthesis. *arXiv preprint arXiv:1809.11096*.

Casanova, A.; Careil, M.; Verbeek, J.; Drozdzal, M.; and Romero Soriano, A. 2021. Instance-conditioned gan. *Advances in Neural Information Processing Systems*, 34: 27517–27529.

Deng, J.; Dong, W.; Socher, R.; Li, L.-J.; Li, K.; and Fei-Fei, L. 2009. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR*.

Dhariwal, P.; and Nichol, A. 2021. Diffusion models beat gans on image synthesis. *Advances in Neural Information Processing Systems*, 34: 8780–8794.

Dinh, L.; Krueger, D.; and Bengio, Y. 2014. Nice: Nonlinear independent components estimation. *arXiv preprint arXiv:1410.8516*.

Dinh, L.; Sohl-Dickstein, J.; and Bengio, S. 2016. Density estimation using real nvp. *arXiv preprint arXiv:1605.08803*.

Esser, P.; Rombach, R.; and Ommer, B. 2021. Taming transformers for high-resolution image synthesis. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 12873–12883.

Gu, J.; Chen, T.; Berthelot, D.; Zheng, H.; Wang, Y.; Zhang, R.; Dinh, L.; Bautista, M. A.; Susskind, J.; and Zhai, S. 2025. STARFlow: Scaling Latent Normalizing Flows for High-resolution Image Synthesis. *arXiv preprint arXiv:2506.06276*.

Gu, J.; Wang, Y.; Zhang, Y.; Zhang, Q.; Zhang, D.; Jaitly, N.; Susskind, J.; and Zhai, S. 2024. Dart: Denoising autoregressive transformer for scalable text-to-image generation. *arXiv preprint arXiv:2410.08159*.

Heusel, M.; Ramsauer, H.; Unterthiner, T.; Nessler, B.; and Hochreiter, S. 2017. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems*, 30.

Ho, J.; Saharia, C.; Chan, W.; Fleet, D. J.; Norouzi, M.; and Salimans, T. 2022. Cascaded Diffusion Models for High Fidelity Image Generation. *J. Mach. Learn. Res.*, 23: 47–1.

Karras, T.; Aittala, M.; Aila, T.; and Laine, S. 2022. Elucidating the design space of diffusion-based generative models. *arXiv preprint arXiv:2206.00364*.

Kingma, D. P.; Salimans, T.; Jozefowicz, R.; Chen, X.; Sutskever, I.; and Welling, M. 2016. Improved variational inference with inverse autoregressive flow. *Advances in neural information processing systems*, 29.

Lee, D.; Kim, C.; Kim, S.; Cho, M.; and Han, W.-S. 2022. Autoregressive image generation using residual quantization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 11523–11532.

Lee, S.-H.; Park, S.; and Kim, G.-M. 2024. REPA-E: End-to-End Training of Latent-Diffusion Models via Representation Alignment. In *arXiv preprint arXiv:2405.18373*.

Li, T.; Tian, Y.; Li, H.; Deng, M.; and He, K. 2024. Autoregressive Image Generation without Vector Quantization. *arXiv preprint arXiv:2406.11838*.

Ma, N.; Goldstein, M.; Albergo, M. S.; Boffi, N. M.; Vanden-Eijnden, E.; and Xie, S. 2024. Sit: Exploring flow and diffusion-based generative models with scalable interpolant transformers. *arXiv preprint arXiv:2401.08740*.

Nash, C.; Menick, J.; Dieleman, S.; and Battaglia, P. W. 2021. Generating Images with Sparse Representations. In *Proceedings of the 38th International Conference on Machine Learning (ICML)*.

Nichol, A. Q.; and Dhariwal, P. 2021. Improved denoising diffusion probabilistic models. In *International Conference on Machine Learning*, 8162–8171. PMLR.

Papamakarios, G.; Murray, I.; and Pavlakou, T. 2017. Masked Autoregressive Flow for Density Estimation. In Guyon, I.; von Luxburg, U.; Bengio, S.; Wallach, H. M.; Fergus, R.; Vishwanathan, S. V. N.; and Garnett, R., eds., *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, 2338–2347.

Peebles, W.; and Xie, S. 2023. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 4195–4205.

Rezende, D.; and Mohamed, S. 2015. Variational Inference with Normalizing Flows. In Bach, F.; and Blei, D., eds., *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, 1530–1538. Lille, France: PMLR.

Rombach, R.; Blattmann, A.; Lorenz, D.; Esser, P.; and Ommer, B. 2022. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 10684–10695.

Salimans, T.; Goodfellow, I.; Zaremba, W.; Cheung, V.; Radford, A.; and Chen, X. 2016. Improved techniques for training gans. *Advances in neural information processing systems*, 29.

Song, Y.; and Dhariwal, P. 2023. Improved techniques for training consistency models. *arXiv preprint arXiv:2310.14189*.

Song, Y.; Dhariwal, P.; Chen, M.; and Sutskever, I. 2023. Consistency Models. *arXiv preprint arXiv:2303.01469*.

Su, J.; Ahmed, M.; Lu, Y.; Pan, S.; Bo, W.; and Liu, Y. 2024. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568: 127063.

Sun, P.; Jiang, Y.; Chen, S.; Zhang, S.; Peng, B.; Luo, P.; and Yuan, Z. 2024. Autoregressive Model Beats Diffusion: Llama for Scalable Image Generation. *arXiv preprint arXiv:2406.06525*.

Tian, K.; Jiang, Y.; Yuan, Z.; Peng, B.; and Wang, L. 2024a. Visual autoregressive modeling: Scalable image generation via next-scale prediction. *arXiv preprint arXiv:2404.02905*.

Tian, Y.; Wang, X.; Li, S.; Wu, Z.; Ye, B.; Zheng, Y.; Liu, B.; Li, J.; and Zhou, J.-R. 2024b. U-REPA: A U-Net based representation alignment framework for accelerating diffusion model training. In *arXiv preprint arXiv:2405.16642*.

Touvron, H.; Lavril, T.; Izacard, G.; Martinet, X.; Lachaux, M.-A.; Lacroix, T.; Rozière, B.; Goyal, N.; Hambro, E.; Azhar, F.; et al. 2023a. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Touvron, H.; Martin, L.; Stone, K.; Albert, P.; Almahairi, A.; Babaei, Y.; Bashlykov, N.; Batra, S.; Bhargava, P.; Bhosale, S.; et al. 2023b. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Tschannen, M.; Pinto, A. S.; and Kolesnikov, A. 2024. JetFormer: An autoregressive generative model of raw images and text. *arXiv preprint arXiv:2411.19722*.

Wang, S.; Gao, Z.; Zhu, C.; Huang, W.; and Wang, L. 2025a. PixNerd: Pixel Neural Field Diffusion. arXiv:2507.23268.

Wang, S.; Li, Z.; Song, T.; Li, X.; Ge, T.; Zheng, B.; and Wang, L. 2024. Exploring DCN-like architecture for fast image generation with arbitrary resolution. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Wang, S.; Tian, Z.; Huang, W.; and Wang, L. 2025b. DDT: Decoupled Diffusion Transformer. arXiv:2504.05741.

Yu, S.; Kwak, S.; Jang, H.; Jeong, J.; Huang, J.; Shin, J.; and Xie, S. 2024. Representation alignment for generation: Training diffusion transformers is easier than you think. *arXiv preprint arXiv:2410.06940*.

Zhai, S.; Zhang, R.; Nakkiran, P.; Berthelot, D.; Gu, J.; Zheng, H.; Chen, T.; Bautista, M. A.; Jaitly, N.; and Susskind, J. 2024. Normalizing flows are capable generative models. *arXiv preprint arXiv:2412.06329*.

Zheng, Y.; Tian, Y.; Li, S.; Wu, Z.; Liu, B.; Li, J.; Ye, B.; and Zhou, J.-R. 2024. LightningDiT: A Vision-Foundation-Model-Aligned VAE for Fast and High-Quality Generation. In *arXiv preprint arXiv:2405.15438*.