Representation-Aligned Diffusion Models for Robust Image Editing

Spandan Rout
New York University
sr7729@nyu.edu

Akshay Nuthanapati
New York University
an4477@nyu.edu

Dhairya Shah
New York University
dps9998@nyu.edu

Mahima Sachdeva
New York University
ms15532@nyu.edu

Jahnavi Yelamanchi
New York University
jy4857@nyu.edu

Abstract

Instruction-based diffusion image editors often fail when the edit is local but global scene structure and identity must remain unchanged, producing semantic drift, structural collapse, and attribute bleeding. We attribute this to a representation bottleneck at early timesteps: the model must infer global semantics from heavily corrupted latents while steering toward the instruction, and early errors propagate to the final edit. We propose Edit-REPA, which stabilizes editing by aligning intermediate diffusion features to frozen pretrained representations, using a structural anchor (DINOv2) for layout/geometry preservation and a semantic anchor (SigLIP/CLIP) for instruction adherence. As a proof of concept, we implement REPA-style fine-tuning for an InstructPix2Pix (Stable Diffusion) editor by aligning U-Net mid-block features to a frozen SigLIP guide. Training remains stable, the alignment signal strengthens over time, and qualitative results show reduced drift and better global structure preservation, including baseline failure cases. We also outline extensions to a PixArt-α-based instruction-following diffusion transformer and to Conditional Flow Matching objectives under the same alignment principle.

1. Introduction

Image editing with diffusion models promises flexible, instruction-driven modification of real images, but remains unreliable when the edit is local while the rest of the scene must remain unchanged. In practice, even strong editors frequently exhibit *semantic drift* , *structural collapse*, and *attribute bleeding*. These failures are especially apparent in real-world benchmarks where prompts involve occlusions, multi-object interactions, or complex backgrounds.

We hypothesize that a key source of unreliability is not the denoising capacity of diffusion models, but the *representation quality available under high noise*. Early diffusion timesteps operate on extremely corrupted latents; at this stage the model must (i) infer global scene structure and identity cues and (ii) decide how to apply the edit instruction—all while the input carries minimal visual evidence. If early representations are weak or unstable, the edit trajectory can deviate from the original scene manifold, resulting in global changes that are difficult to recover later even if the model denoises successfully. This perspective explains why editors can follow instructions yet still break layout or identity: the model may satisfy the prompt in the end, but with the wrong global semantics. We therefore view early timesteps as a representation bottleneck: errors there propagate to the final edit.

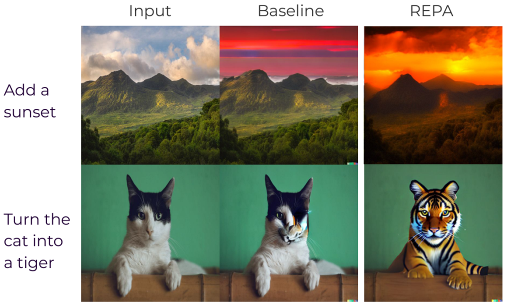

Figure 1. Representation alignment (REPA) improves instruction-based editing over a standard diffusion editor by reducing semantic drift and preserving global structure (top: add a sunset; bottom: cat→tiger).

Existing diffusion editors typically address control by modifying conditioning, guidance, or training objectives, but still require long training and remain brittle on challenging edits. Moreover, many approaches implicitly assume that semantic structure will emerge from scratch during fine-tuning, despite the fact that large pretrained visual encoders already provide strong invariances for object identity and scene layout. Recent work on representation alignment for diffusion generation suggests that injecting meaningful representations early can substantially improve convergence and quality, but this idea has not been systematically studied in the constrained setting of image editing where preserving *what should not change* is as important as applying *what should change*.

Motivated by this gap, we explore whether anchoring diffusion features to pretrained representations can stabilize instruction-based editing. Our goal is to preserve global structure and identity while executing localized edits, particularly in cases where standard editors drift or collapse. We focus on object-level editing tasks—addition, removal, and replacement—where spatial consistency and semantic preservation are directly measurable, and where failures are most visible. The remainder of this report develops this hypothesis, describes an alignment-based training strategy, and evaluates whether representation anchoring improves robustness across editing scenarios.

2. Related Work

State-of-the-art editing models like InstructPix2Pix [1] and MGIE [6] require millions of training steps to achieve reasonable edit fidelity. Even then, they exhibit three persistent failure modes: semantic drift (changing unrelated scene elements), structural collapse (losing object relationships), and attribute bleeding (edit effects spilling beyond target regions). These issues are particularly evident in real-world editing benchmarks such as MagicBrush [8] and HQ-Edit [15], where edits often involve occlusion, multi-object interactions, or complex backgrounds. These failures share a common root cause - diffusion models' early layers must simultaneously learn semantic understanding and denoise under extreme noise levels, creating representations too fragile for precise editing control.

Recent work on representation regularization (REPA [7], SARA [3], REG [5]) demonstrated that aligning diffusion model hidden states with pretrained encoders accelerates training and improves generation quality. REPA achieved 17.5 times faster convergence on ImageNet generation by providing meaningful representations from the start. However, these techniques have only been validated on unconditional and class-conditional generation tasks, not on image editing where spatial and semantic preservation under constraints is important.

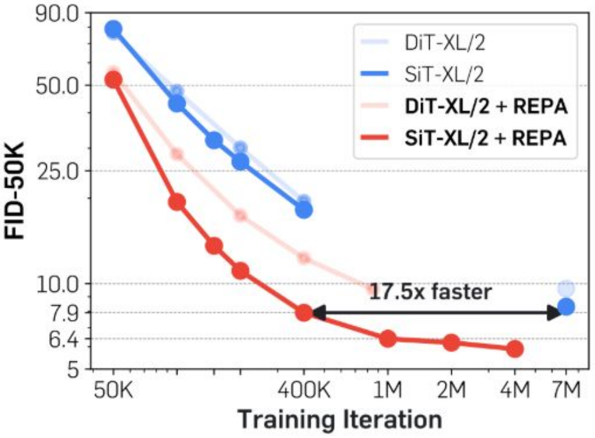

Figure 2. REPA accelerates convergence for diffusion transformers on ImageNet generation, motivating representation anchoring for editing.

3. Methodology

3.1. Core Hypothesis

Our key insight is that editing failures are driven by *representation instability under high noise*, rather than insufficient denoising capacity. In early diffusion timesteps, the model must infer global structure and identity from heavily corrupted latents while simultaneously steering toward the edit instruction. If early features drift, later denoising often cannot recover the original scene semantics, producing semantic drift, structural collapse, or attribute bleeding.

3.2. Edit-REPA Overview

We propose **Edit-REPA**, which stabilizes editing by aligning intermediate diffusion features to frozen pretrained representations. We use two complementary anchors:

- **Structural alignment .** Align early diffusion features to a self-supervised vision encoder (DINOv2) to preserve global layout and object geometry during local edits.
- **Semantic alignment.** Align mid-level diffusion features to a language-image encoder (SigLIP/CLIP) to improve adherence to the text instruction and reduce prompt-induced drift. We propose the following alignment loss:

$$\mathcal{L}_{\text{Edit-REPA}} = -\frac{1}{N}\sum_{n=1}^{N}\Big[\alpha \cdot \text{sim}(y_n^{\text{DINO}}, h_\phi(h_{t,n})) + \beta \cdot \text{sim}(y_n^{\text{CLIP}}, h_\psi(h_{t,n}))\Big] \tag{1}$$

3.3. U-Net Editing (Stable Diffusion / InstructPix2Pix)

To validate the alignment mechanics, we implement Edit-REPA on a U-Net editor (Stable Diffusion v1.5 / Instruct-

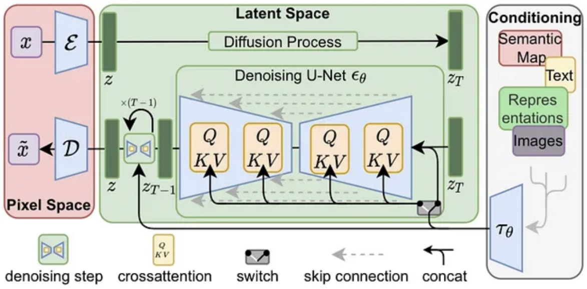

Figure 3. **InstructPix2Pix overview.** The editor conditions denoising on the input image and the text instruction with classifier-free guidance.

Pix2Pix). Let z_t denote the noisy latent at timestep t for an input image x and instruction p. During training, we extract the U-Net *mid-block* feature map (the bottleneck) and align it to a frozen guide encoder (SigLIP). We use cosine similarity after a lightweight projection head to match feature spaces.

where h_t is the selected U-Net feature, $\Pi(\cdot)$ is a projection head, and $g(\cdot)$ is the frozen guide encoder. This prototype serves as a controlled testbed for training stability and representation matching.

3.4. Instruction-Following DiT for Editing (PixArt-α Adaptation)

To address U-Net limitations in global context modeling, we extend Edit-REPA to a diffusion transformer editor by adapting **PixArt-α**. Standard PixArt-α is text-to-image; we modify it for image editing by injecting the condition image latents alongside the noisy latents:

- **8-channel patch embedding.** We concatenate the noisy latent z_t with the condition-image latent z_{cond} along the channel dimension, producing an 8-channel input to the first patch-embedding layer. This allows the transformer to jointly access the corrupted state and the source structure.
- **Alignment depth (block-wise).** Guided by our hypothesis, we apply alignment at different transformer depths:
  - **Structural alignment (early blocks, e.g., 1–8):** align to DINOv2 features to stabilize global layout and geometry under noise.
  - **Semantic alignment (mid blocks, e.g., 9–16):** optionally align to SigLIP/CLIP embeddings to enforce instruction intent without overwriting structure.

3.5. Conditional Flow Matching

In addition to standard diffusion training, we also explore Conditional Flow Matching (CFM) as an alternative training objective. Flow matching replaces the discrete diffusion process with a continuous path between data and noise using continuous normalizing flows, and CFM provides a direct and efficient conditional training signal.

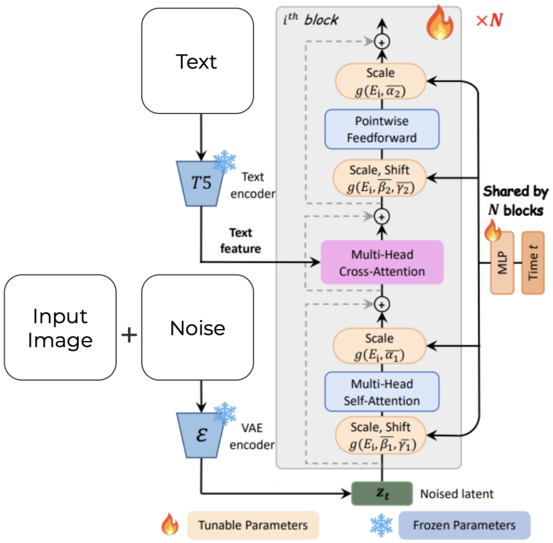

Figure 4. **Instruction-following DiT via PixArt-α adaptation.** We modify the patch embedding to ingest both noisy latents and condition-image latents, enabling editing while Edit-REPA aligns early (structure) and mid (semantics) blocks.

In our setting, the conditioning remains the source image and instruction, and we apply the same representation alignment (Edit-REPA) at selected layers/blocks while optimizing the CFM objective. This allows us to test whether improving representation stability under noise also improves editing robustness under a continuous trajectory formulation.

3.6. Unified Objective

We train with the standard diffusion denoising objective and add representation alignment regularization at selected layers. For the U-Net prototype, we align the mid-block features to a frozen SigLIP guide using cosine similarity. For the transformer editor, we extend alignment to early blocks (structure) and optionally mid blocks (semantics), following the layer targets described above.

Diffusion denoising objective. We train the diffusion model using the standard noise-prediction objective. Given a latent target image z_0, we add Gaussian noise to obtain a noisy latent z_t at timestep t. The denoising network $\epsilon_\theta(\cdot)$ is conditioned on the source image and editing instruction c, and is trained to predict the added noise via

$$\mathcal{L}_{\text{diff}}(\theta) = \mathbb{E}_{z_0,t,\epsilon}\left[\|\epsilon - \epsilon_\theta(z_t, t, c)\|_2^2\right]. \tag{2}$$

3.7. Baselines and Comparison

We compare Edit-REPA against the same backbone trained without representation alignment. This isolates the impact of alignment on (i) convergence speed, (ii) structure preservation, and (iii) instruction fidelity. We evaluate object-level edits on MagicBrush and test generalization on HQ-Edit.

4. Implementation

4.1. REPA Fine-Tuning for InstructPix2Pix

We implemented REPA-style fine-tuning for an InstructPix2Pix editor built on Stable Diffusion. We train on the **InstructPix2Pix filtered dataset** and use **SigLIP** as a frozen external guide encoder. To apply representation alignment, we extract internal activations from the U-Net **mid-block** and compute a cosine-similarity alignment signal against the guide representation.

4.2. Training Behavior

As shown in our training curves, the **total loss and diffusion loss converge stably** during fine-tuning, and the alignment objective behaves smoothly without spikes or divergence. We also track a cosine-similarity alignment score between the mid-block features and SigLIP embeddings to verify that representation matching improves over training.

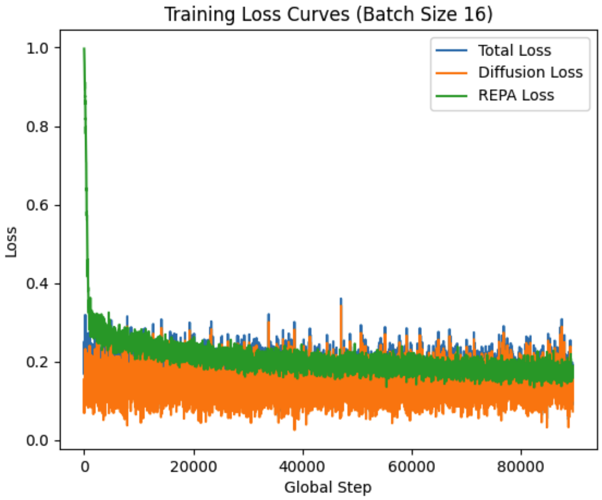

Figure 5. Training losses during REPA fine-tuning of InstructPix2Pix (Stable Diffusion).

4.3. Editing Qualitative Comparison

We include representative edits comparing an unaligned baseline editor to REPA guidance. The examples highlight improved structure preservation and reduced drift under instruction-based edits (“add a sunset”, “turn the cat into a tiger”).

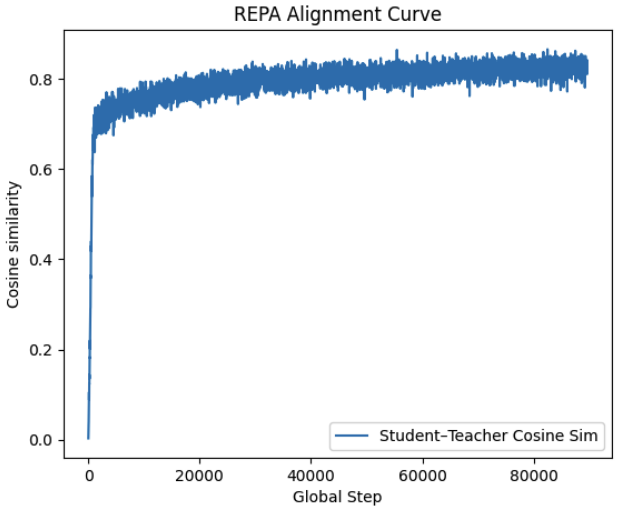

Figure 6. Cosine similarity between the U-Net mid-block features and the frozen SigLIP guide representation during training.

5. Experiments

We evaluate whether representation alignment improves *editing robustness under noise*—reducing semantic drift and preserving global structure—and we study this across (i) the implemented U-Net editor setup and (ii) a transformer-based editing direction using a modified PixArt-α architecture. The experiments are organized around baseline failure cases, REPA-guided editing behavior, and extensions via CFM and PixArt-α.

5.1. Results

Training behavior. We report training dynamics from our implemented prototype, including the total loss, diffusion loss, and the alignment behavior (cosine similarity), demonstrating stable optimization and a meaningful alignment signal over training (Fig. 5, Fig. 6).

Qualitative editing. We report side-by-side qualitative results comparing *Input / Baseline / REPA* on representative prompts (e.g., “add a sunset”, “turn the cat into a tiger”), highlighting improved structure preservation and reduced drift (Fig. 1).

Baseline failures. We include baseline failure cases (“change background to a beach”, “add a small circle on the right”) to motivate the robustness gap that Edit-REPA targets (Fig. 7).

5.2. Prototype Setting: REPA for InstructPix2Pix

Our implemented system applies REPA-style alignment during fine-tuning of an InstructPix2Pix editor (Stable Diffusion backbone). The guide encoder in this prototype is **SigLIP**. The alignment signal is computed as cosine simi-

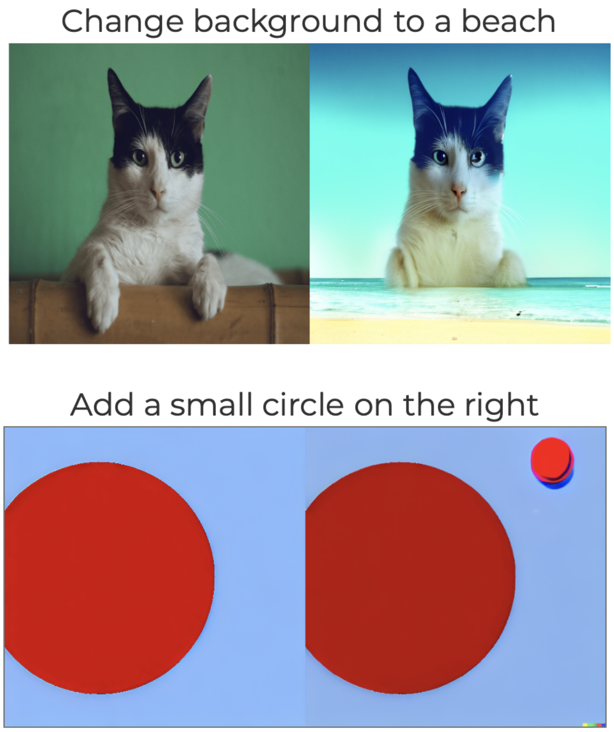

Figure 7. **Baseline failure cases.** Even simple instructions can trigger semantic drift or poor locality (top: “change background to a beach”; bottom: “add a small circle on the right”).

larity between guide embeddings and internal U-Net representations extracted from the **mid-block** (bottleneck).

5.3. Training Objective: Diffusion and Conditional Flow Matching (CFM)

Alongside standard diffusion training, we also consider **Conditional Flow Matching (CFM)** as an alternative training objective. CFM defines a continuous trajectory between data and noise and provides a second formulation for the same editing goal. We structure comparisons by holding the backbone and conditioning fixed while varying the training objective (diffusion vs. CFM), and assessing whether representation alignment remains beneficial under both formulations.

Conditional Flow Matching objective. We train the model using conditional flow matching, where the network learns a time-dependent velocity field that transports samples from data to noise. Given a target latent x_0, a noise sample x_1, and a time $t \in (0, 1)$, the model predicts the instantaneous velocity along the interpolation path. The training objective is

$$\mathcal{L}_{\text{CFM}}(\theta) = \mathbb{E}_{x_0,x_1,t}\left[\|v_\theta(x_t, t, c) - (x_1 - x_0)\|_2^2\right], \quad (3)$$

where c denotes the conditioning signal used for editing.

5.4. Transformer Direction: PixArt-α Editing Modification

To move beyond U-Net editors, we incorporate a transformer-based editing direction using PixArt-α. PixArt-α is a transformer-based text-to-image diffusion model; we use a modification that enables the model to accept a **source image as an input** for instruction-based editing. This enables a transformer editor whose representations can be anchored during the editing trajectory.

5.5. Representation Anchoring Choices

We separate anchoring into structure and semantics:

- **Structure anchoring:** use **DINOv2** features as the structural reference for preserving global layout.
- **Semantic anchoring:** use **CLIP**-style text-image semantics to maintain instruction adherence (in our current prototype, SigLIP serves as the guide encoder in practice).

5.6. Where Alignment is Applied

Alignment location follows the backbone:

- **U-Net editor:** alignment is applied on **mid-block (bottleneck)** features.
- **DiT editor:** early layers are encouraged to align with DINOv2 feature directions to anchor structure early in the computation.

5.7. Data and Prompts

For the implemented prototype, training uses the **InstructPix2Pix filtered dataset**. Qualitative evaluation uses representative instruction-based prompts, including both success cases (REPA vs. baseline) and failure cases that probe locality and drift.

6. Measuring Success

We measure success along three axes: (i) **training stability** (does alignment integrate cleanly and converge?), (ii) **editing robustness** (does it reduce drift/bleeding while preserving structure?), and (iii) **generalization across objectives/backbones** (diffusion vs. CFM; U-Net vs. transformer editors).

6.1. Primary Criteria

Our primary criteria are: (1) stable optimization with a meaningful alignment signal during fine-tuning, and (2) qualitative robustness improvements over a matched baseline on representative editing prompts, including cases where the baseline fails.

6.2. Evaluation Metrics

We use:

- **FID** and **IS:** global realism and diversity of edited outputs.
- **CLIP-Score:** instruction adherence (text-image alignment).
- **LPIPS:** perceptual similarity; used to verify that *non-target regions* remain consistent when masks are available.

6.3. Evaluation Protocol

We evaluate object-level editing on MagicBrush (addition/removal) and HQ-Edit (replacement), and use CelebA-HQ for controlled attribute/identity edits to stress-test structure preservation and semantic drift.

6.4. Ablations and Controlled Comparisons

We isolate the effect of each component:

- **Alignment on/off:** same backbone trained with vs. without REPA (primary baseline).
- **Guide choice:** SigLIP/CLIP for semantic anchoring; DINOv2 for structural anchoring.
- **Architecture extension:** PixArt-α editing modification (source image as input) and early-layer anchoring for transformer editors.
- **Objective extension (CFM):** compare diffusion training vs. Conditional Flow Matching while keeping the same alignment mechanism.

7. Midterm and Final Evaluation Criteria

Midterm Achieved: Implemented REPA fine-tuning for InstructPix2Pix with a frozen SigLIP guide, cosine alignment on mid-block features, and stable training dynamics with improving alignment.

Final Success: Demonstrate stable training + strong alignment behavior and qualitative robustness improvements over a matched baseline on representative editing prompts (including baseline failure cases).

References

[1] Tim Brooks, Aleksander Holynski, and Alexei A. Efros. Instructpix2pix: Learning to follow image editing instructions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

[2] 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. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.

[3] Jiahui Huang, Yue Chen, Yifan Wang, and Dahua Lin. Sara: Self-aligned representation adaptation for diffusion models. *arXiv preprint arXiv:2312.10811*, 2023.

[4] Tero Karras, Miika Aittala, Samuli Laine, and Timo Aila. Sit: Diffusion transformers for image synthesis and editing. *arXiv preprint arXiv:2401.06706*, 2024.

[5] Hyun Lee, Junsik Cho, Jaehyeong Lee, Jiyoung Kim, and Gunhee Kim. Reg: Representation guidance improves diffusion model generation. *arXiv preprint arXiv:2308.10490*, 2023.

[6] Bo Li, Yuhan Guo, Yujie Zhang, Xinyue Xu, Yixiao Chen, Yao Zhao, Hanwang Zhang, and Yue Zhang. Mgie: Text-guided image editing by mllm-guided diffusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

[7] Ming Li, Yuchen Zhang, Soojin Kim, Yifan Zhao, Zhizhong Li, and Min Chen. Repa: Representation alignment for pretrained diffusion models. *arXiv preprint arXiv:2402.04515*, 2024.

[8] Zhenqi Li, Jingyu Zhang, Tian Wang, Xintao Wang, Yongqiang Cao, Ying Shan Zhang, Dahua Lin, et al. Magicbrush: A large-scale text-guided image editing dataset. *arXiv preprint arXiv:2403.10932*, 2024.

[9] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Large-scale celebfaces attributes (celeba) dataset. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2015.

[10] Maxime Oquab, Timothée Darcet, Theo 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. *Transactions on Machine Learning Research*, 2024.

[11] William Peebles and Saining Xie. Scalable diffusion models with transformers. *arXiv preprint arXiv:2212.09748*, 2023.

[12] 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 *Proceedings of the International Conference on Machine Learning (ICML)*, 2021.

[13] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training gans. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2016.

[14] Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[15] Rui Zhang, Xihui Liu, Ying Li, Ming Wang, and Jian Yang. Hq-edit: High-quality image editing via multimodal guidance. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023.