# Dual-Path Condition Alignment for Diffusion Transformers

*Changhao Peng*[†,1], *Wei Gao*[†,2]

[†]*Peking University, Shenzhen, China*

1 ✉ *pch@stu.pku.edu.cn* 2 ✉ *gaowei262@pku.edu.cn*

北京大学信息工程学院

School of Electronic and Computer Engineering

Peking University

## Abstract

Denoising-based generative models have been significantly advanced by representation-alignment (REPA) loss, which leverages pre-trained visual encoders to guide intermediate network features. However, REPA's reliance on external visual encoders introduces two critical challenges: potential *distribution mismatches* between the encoder's training data and the generation target, and the high *computational costs* of pre-training. Inspired by the observation that REPA primarily aids early layers in capturing robust semantics, we propose an unsupervised alternative that avoids external visual encoder and the assumption of consistent data distribution. We introduce ***DU**al-**P**ath condition **A**lignment* (**DUPA**), a novel self-alignment framework, which independently noises an image multiple times and processes these noisy latents through decoupled diffusion transformer, then aligns the derived conditions—low-frequency semantic features extracted from each path. Experiments demonstrate that DUPA achieves FID=1.46 on ImageNet 256x256 with only 400 training epochs, outperforming all methods that do not rely on external supervision. DUPA is also model-agnostic and can be readily applied to any denoising-based generative model, showcasing its excellent scalability and generalizability.

## Introduction

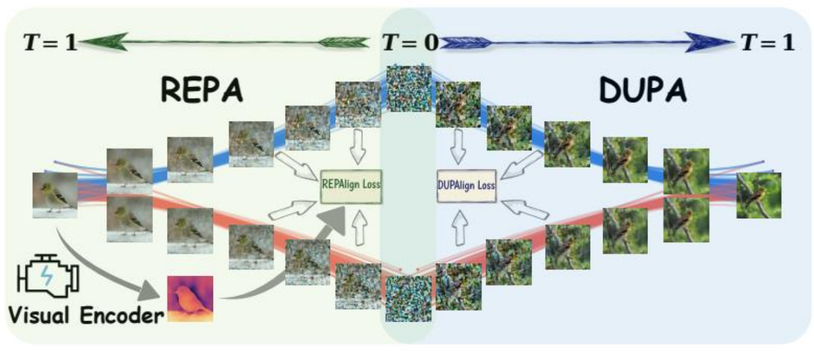



Figure 2: **Comparison between REPA and DUPA.** REPA needs an external visual encoder to generate effective representations, whereas DUPA can get effective representations through internal alignment.

As illustrated on the left of Figure, REPA acts like a "data annotator" during training, supplying "labels" (i.e., effective representations) obtained from "ground truth"(i.e., pure images) for noisy images, which is similar to supervised learning. However, as discussed above, this "supervised learning" approach in REPA faces two challenges compared to unsupervised learning: "costliness of labeling" and "inaccurate labeling" issues.

## Method

**Algorithm 1** Dual-Path Condition Alignment Batch Step

**Input:** DDT $v_\theta$, batch of $B$ flow examples $F = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_B, y_B)\}$, projector $z_\phi$, learning rate $\beta$, sampling times $K = 2$ and hyperparameter $\lambda = 0.5$.
**Output:** Updated model parameters $\theta$.
$L(\theta, \phi) = 0$
**for** $i$ in range$(B)$ **do**
  **for** $j$ in range$(K)$ **do**
    $t_j \sim U(0,1)$, $\epsilon_j \sim \mathcal{N}(0, \mathrm{I})$, $\mathbf{x}_{t_j} = \alpha_{t_j}\mathbf{x}_i + \sigma_{t_j}\epsilon_j$
    $\hat{\mathbf{v}}_j, \mathbf{z}_j = v_\theta(\mathbf{x}_{t_j}, t_j, y_i)$, $\mathbf{v}_j = \dot{\alpha}_{t_j}\mathbf{x}_i + \dot{\sigma}_{t_j}\epsilon_j$
    $\mathbf{z}_j = z_\phi(\mathbf{z}_j)$
    $L(\theta, \phi) += \|\hat{\mathbf{v}}_j - \mathbf{v}_j\|^2$
    **for** $k$ in range$(j)$ **do**
      $L(\theta, \phi) -= \frac{2\lambda}{K(K-1)} \cdot \mathrm{sim}(\mathbf{z}_k, \mathbf{z}_j)$
    **end for**
  **end for**
**end for**
$\theta \leftarrow \theta - \frac{\beta}{B}\nabla_\theta L(\theta, \phi)$, $\phi \leftarrow \phi - \frac{\beta}{B}\nabla_\phi L(\theta, \phi)$

An image is independently noised multiple times during training, and use Decoupled Diffusion Transformer to predict different denoising paths. In this way, the condition encoder can extract different conditions, which are low-frequency semantic features from different noisy images. Since these conditions originate from the same pure image, they should be similar, much like the representations obtained by large visual encoders in REPA. We propose to align these different conditions derived from independently noised versions of a single image to furnish effective representation guidance for model training.

## Results

Table 1: **System-Level Performance on ImageNet 256 × 256.** Our results are **bolded** to indicate that DUPA performs better than methods without external supervision of large visual encoders, while highlighted to indicate that DUPA performs the best among all methods. ↓ indicates a lower value is better and ↑ indicates a higher value is better.

| Method | Training Epochs | #params | External Images | External Params | Generation w/o CFG | | | | | Generation w/ CFG | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | FID↓ | sFID↓ | IS↑ | Prec.↑ | Rec.↑ | FID↓ | sFID↓ | IS↑ | Prec.↑ | Rec.↑ |
| **No Auxiliary Task** | | | | | | | | | | | | | | |
| DiT | 1400 | 675M | 0 | 0 | 9.62 | 6.85 | 121.5 | 0.67 | 0.67 | 2.27 | 4.60 | 278.2 | 0.83 | 0.57 |
| SiT | 1400 | 675M | 0 | 0 | 8.61 | 6.32 | 131.7 | 0.68 | 0.67 | 2.06 | 4.50 | 270.3 | 0.82 | 0.59 |
| FasterDiT | 400 | 675M | 0 | 0 | 7.91 | 5.45 | 131.3 | 0.67 | **0.69** | 2.03 | 4.63 | 264.0 | 0.81 | 0.60 |
| DDT | 400 | 675M | 0 | 0 | 8.06 | 5.31 | 127.4 | 0.69 | 0.67 | 2.01 | 4.66 | 281.7 | 0.80 | 0.59 |
| **Masked Image Modeling** | | | | | | | | | | | | | | |
| MaskGIT | 555 | 227M | 0 | 0 | 6.18 | - | 182.1 | **0.80** | 0.51 | - | - | - | - | - |
| LlamaGen | 300 | 3.1B | 0 | 0 | 9.38 | 8.24 | 112.9 | 0.69 | 0.67 | 2.18 | 5.97 | 263.3 | 0.81 | 0.58 |
| VAR | 350 | 2.0B | 0 | 0 | - | - | - | - | - | 1.80 | - | 365.4 | 0.83 | 0.57 |
| MagViT-v2 | 1080 | 307M | 0 | 0 | 3.65 | - | 200.5 | - | - | 1.78 | - | 319.4 | - | - |
| MAR | 800 | 945M | 0 | 0 | **2.35** | - | **227.8** | 0.79 | 0.62 | 1.55 | - | 303.7 | 0.81 | 0.62 |
| MaskDiT | 1600 | 675M | 0 | 0 | 5.69 | 10.34 | 177.9 | 0.74 | 0.60 | 2.28 | 5.67 | 276.6 | 0.80 | 0.61 |
| MDT | 1300 | 675M | 0 | 0 | 6.23 | 5.23 | 143.0 | 0.71 | 0.65 | 1.79 | 4.57 | 283.0 | 0.81 | 0.61 |
| MDTv2 | 920 | 675M | 0 | 0 | - | - | - | - | - | 1.58 | 4.52 | 314.7 | 0.79 | 0.65 |
| **Contrastive Learning** | | | | | | | | | | | | | | |
| ΔFM | 800 | 675M | 0 | 0 | - | - | - | - | - | 1.97 | 4.53 | 268.4 | 0.79 | 0.65 |
| Disp-Loss | 1200 | 675M | 0 | 0 | - | - | - | - | - | 1.97 | 4.61 | 275.2 | 0.80 | 0.63 |
| **Supervised Representation Alignment** | | | | | | | | | | | | | | |
| REPA | 80 | 675M | 142M | 1.1B | 7.90 | 5.06 | 122.6 | 0.70 | 0.65 | - | - | - | - | - |
| | 200 | | | | 6.40 | - | - | - | - | 1.96 | 4.49 | 264.0 | 0.82 | 0.60 |
| | 800 | | | | 5.90 | 5.73 | 157.8 | 0.70 | **0.69** | 1.42 | 4.70 | 305.7 | 0.80 | 0.65 |
| **Unsupervised Representation Alignment** | | | | | | | | | | | | | | |
| **DUPA (Ours)** | 80 | 675M | 0 | 0 | 8.71 | 4.65 | 114.6 | 0.70 | 0.65 | 2.28 | 4.48 | 237.2 | 0.83 | 0.59 |
| | 200 | | | | 6.57 | 4.63 | 136.5 | 0.70 | 0.68 | 1.70 | 4.45 | 265.3 | 0.83 | 0.61 |
| | 400 | | | | 5.92 | 4.63 | 149.6 | 0.71 | 0.69 | **1.46** | 4.45 | 296.2 | 0.84 | 0.62 |

Table 2: **Component-wise analysis.** All models are DUPA-L/2 trained for 400K iterations with different settings. "Resampling" column indicates whether to independently resample timestamp $t$ or noise $\epsilon$.

| Resampling | Depth | Objective | λ | FID↓ |
|---|---|---|---|---|
| Vanilla SiT-L/2 | | | | 18.8 |
| $t$ | 8 | Cos. sim. | 0.5 | 13.2 |
| $\epsilon$ | 8 | Cos. sim. | 0.5 | 12.4 |
| $t, \epsilon$ | 4 | Cos. sim. | 0.5 | 11.8 |
| $t, \epsilon$ | 6 | Cos. sim. | 0.5 | 11.3 |
| $t, \epsilon$ | 10 | Cos. sim. | 0.5 | 11.2 |
| $t, \epsilon$ | 12 | Cos. sim. | 0.5 | 11.6 |
| $t, \epsilon$ | 14 | Cos. sim. | 0.5 | 11.9 |
| $t, \epsilon$ | 16 | Cos. sim. | 0.5 | 12.1 |
| $t, \epsilon$ | 8 | NT-Xent | 0.5 | 11.6 |
| $t, \epsilon$ | 8 | Cos. sim. | 0.25 | 11.2 |
| $t, \epsilon$ | 8 | Cos. sim. | 0.75 | 11.1 |
| $t, \epsilon$ | 8 | Cos. sim. | 1 | 11.1 |
| $t, \epsilon$ | 8 | Cos. sim. | 0.5 | 11.1 |

Table 3: Model performance across different sizes with 400K training steps.

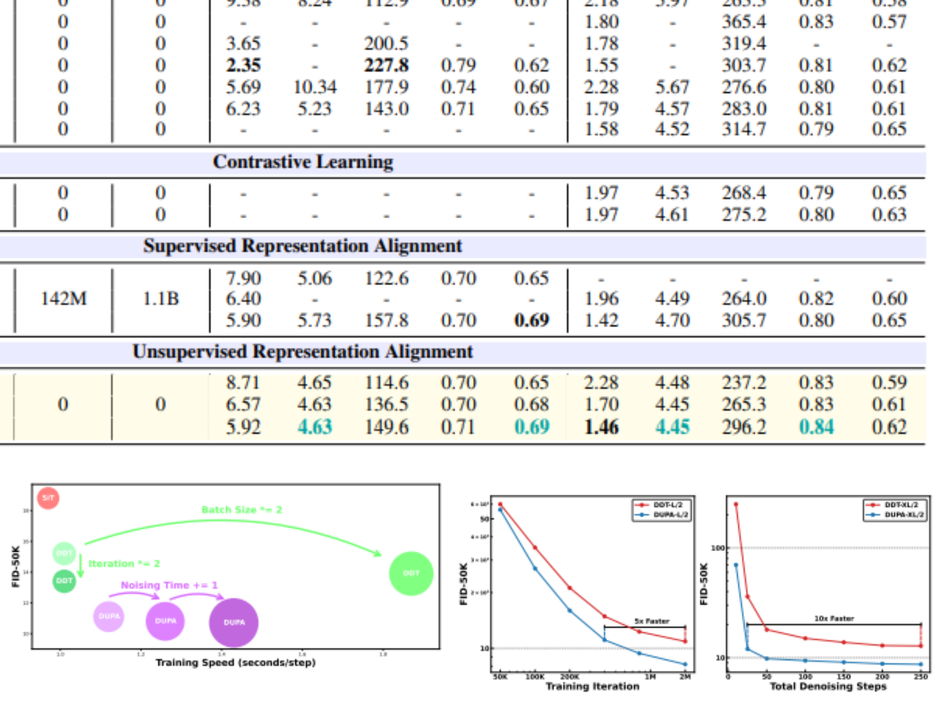
| Model | FID↓ | sFID↓ | IS↑ | Prec.↑ | Rec.↑ |
|---|---|---|---|---|---|
| SiT-B/2 | 33.0 | 6.46 | 43.7 | 0.53 | 0.63 |
| DDT-B/2 | 29.5 | 6.23 | 51.7 | 0.57 | 0.63 |
| **DUPA-B/2** | **25.2** | **5.89** | **67.4** | **0.61** | **0.63** |
| SiT-L/2 | 18.8 | 5.29 | 72.0 | 0.64 | 0.64 |
| DDT-L/2 | 14.9 | 5.17 | 87.8 | 0.65 | 0.64 |
| **DUPA-L/2** | **11.1** | **4.91** | **104.8** | **0.69** | **0.65** |
| SiT-XL/2 | 17.2 | 5.07 | 76.5 | 0.65 | 0.63 |
| DDT-XL/2 | 12.8 | 4.98 | 91.3 | 0.67 | 0.63 |
| **DUPA-XL/2** | **8.71** | **4.65** | **114.6** | **0.70** | **0.65** |

Table 4: Ablation study of proposed improvements.

| Method | FID↓ | sFID↓ | IS↑ | Prec.↑ | Rec.↑ |
|---|---|---|---|---|---|
| DDT-L/2 | 14.9 | 5.17 | 87.8 | 0.65 | 0.64 |
| + *Dual-Path Sampling* | 12.5 | 5.02 | 96.6 | 0.68 | 0.65 |
| + *Condition Alignment* | 11.1 | 4.91 | 104.8 | 0.69 | 0.65 |

| Method | Iter. | BS | K | TS↓ | Mem.↓ | FID↓ |
|---|---|---|---|---|---|---|
| SiT-L/2 | 400K | 256 | 1 | 0.97 | 22.6 | 18.8 |
| DDT-L/2 | 400K | 256 | 1 | 1.01 | 23.3 | 15.2 |
| DDT-L/2 | 400K | 512 | 1 | 1.87 | 35.5 | 13.9 |
| DDT-L/2 | 800K | 256 | 1 | 1.01 | 23.3 | 13.4 |
| DUPA-L/2 | 400K | 256 | 2 | 1.12 | 27.9 | 11.1 |
| DUPA-L/2 | 400K | 256 | 3 | 1.26 | 32.5 | 10.8 |
| DUPA-L/2 | 400K | 256 | 4 | 1.43 | 38.2 | 10.7 |

(a) "BS" indicates batch size, "K" indicates noising times, "TS" indicates training speed (sec/step) and "Mem." indicates memory usage of a single GPU (GB).




(b) Image sampling is performed on DUPA-XL/2 and DDT-XL/2 trained for 400K iterations.

Figure 3: **Time and computational cost analysis.** (a) Time and computational costs comparison (b)Training efficiency and inference speed comparison.