

## A Additional Results

### A.1 Gradient Angle

We provide detailed results of cosine similarity between REPA [19] and denoising gradients. In Figure 1, we separately compute gradients of the feature alignment and the denoising objective for SiT-XL/2 [12] and compare the cosine similarity of their directions at different training iterations. Specifically, we randomly sample 960 images from the training dataset of ImageNet [1] for the comparison and take gradients of parameters in the eighth block of SiT-XL/2 for example (REPA sets the default alignment depth as 8).

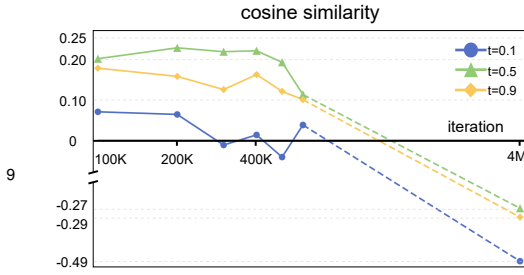


Figure 1: Gradient cosine similarity between REPA and the denoising objective.

| iteration | t = 0.02 | t = 0.04 | t = 0.06 | t = 0.08 | t = 0.10 |
|-----------|----------|----------|----------|----------|----------|
| 100K      | 0.0070   | 0.0064   | 0.0327   | 0.0525   | 0.0692   |
| 200K      | 0.0350   | 0.0476   | 0.0434   | 0.0568   | 0.0628   |
| 300K      | -0.0235  | -0.0324  | -0.0316  | -0.0044  | -0.0116  |
| 400K      | -0.1236  | -0.1056  | -0.1133  | 0.0232   | 0.0130   |
| 500K      | 0.0346   | -0.0368  | -0.0246  | -0.0063  | -0.0409  |
| 600K      | -0.1185  | -0.0546  | 0.0645   | -0.0039  | 0.0372   |
| 4M        | -0.2065  | -0.1279  | -0.1928  | -0.3621  | -0.4942  |

Table 1: Detailed cosine similarity results of the 8<sup>th</sup> block in SiT-XL/2 at  $t \leq 0.10$ .

We first observe a relatively high cosine similarity, representing an acute angle between gradients of the two objectives. However, the similarity shows a decreasing trend as the training progresses, and the angle becomes nearly orthogonal at the intermediate stage (around 400K iteration). Furthermore, we find that the similarity becomes obviously negative at the final training stage, such as at 4M iteration, indicating that there might be some potential conflict between REPA and diffusion loss.

In addition to training iterations, we also find a feature alignment gap over different diffusion timesteps: As reported in [19], a well-trained DiT [15] or SiT exhibits a higher feature alignment at the intermediate diffusion timesteps, while the alignment is notably weaker at those closer to the data distribution, i.e., nearby the sampling results, such as  $t = 0.1$  for SiT. We observe a similar trend in our gradient similarity comparison. According to diffusion sampling properties, the initial steps starting from noise mainly contribute to global fidelity, namely the basic outline of images, while the steps closer to the data are to refine microscopic details such as textures [6]. We hypothesize that the diffusion transformer eventually needs to refine its own representations for detail generation beyond learning directly from external features.

| iteration | t = 0.02 | t = 0.05 | t = 0.07 | t = 0.1 | t = 0.2 | t = 0.5 | t = 0.9 |
|-----------|----------|----------|----------|---------|---------|---------|---------|
| 100K      | -0.0138  | -0.0131  | -0.0068  | 0.0129  | 0.0488  | 0.0541  | -0.0093 |
| 200K      | -0.0423  | -0.0674  | -0.0719  | -0.0491 | 0.0068  | 0.0801  | 0.0099  |
| 250K      | -0.0323  | -0.0597  | -0.0598  | -0.0599 | -0.0264 | 0.0354  | 0.0419  |
| 260K      | -0.0232  | -0.0331  | -0.0243  | -0.0034 | 0.0436  | 0.0729  | 0.0065  |
| 270K      | 0.0029   | 0.0152   | 0.0113   | 0.0097  | 0.0419  | 0.0554  | 0.0233  |
| 280K      | -0.0263  | -0.0131  | -0.0031  | 0.0011  | 0.0217  | 0.0455  | -0.0176 |
| 290K      | -0.0524  | 0.0199   | 0.0308   | 0.0532  | 0.0832  | 0.0550  | 0.0111  |

Table 2: Detailed gradient cosine similarity results between holistic alignment and denoising objectives on the 8<sup>th</sup> block of SiT-XL/2 at different training iterations.

For our method, HASTE, we also examine the gradient cosine similarity between holistic alignment and denoising. The similarity trend serves as a kind of reference for our termination strategy.

## 26 A.2 Detailed Quantitative Results

27 We provide detailed evaluation results of HASTE on different SiT models in Table 3. All results are  
 28 reported with the SDE Euler-Maruyama sampler (NFEs = 250) and without classifier-free guidance.

| model         | #params | iteration | FID↓ [3]    | sFID↓ [13]  | IS↑ [18]     | Prec.↑ [8]  | Rec.↑ [8]   |
|---------------|---------|-----------|-------------|-------------|--------------|-------------|-------------|
| SiT-B/2 [12]  | 130M    | 400K      | 33.0        | 6.46        | 43.7         | 0.53        | 0.63        |
| <b>+HASTE</b> | 130M    | 100K      | 39.9        | 7.16        | 35.8         | 0.52        | 0.61        |
| <b>+HASTE</b> | 130M    | 200K      | 25.7        | 6.66        | 57.0         | 0.59        | 0.62        |
| <b>+HASTE</b> | 130M    | 400K      | <b>19.6</b> | <b>6.38</b> | <b>73.0</b>  | <b>0.62</b> | <b>0.64</b> |
| SiT-L/2 [12]  | 458M    | 400K      | 18.8        | 5.29        | 72.0         | 0.64        | 0.64        |
| <b>+HASTE</b> | 458M    | 100K      | 19.6        | 5.70        | 67.9         | 0.64        | 0.63        |
| <b>+HASTE</b> | 458M    | 200K      | 12.1        | 5.28        | 96.1         | 0.68        | 0.64        |
| <b>+HASTE</b> | 458M    | 400K      | <b>8.9</b>  | <b>5.18</b> | <b>118.9</b> | <b>0.69</b> | <b>0.66</b> |
| SiT-XL/2 [12] | 675M    | 7M        | 8.6         | 6.32        | 131.7        | 0.68        | 0.67        |
| <b>+HASTE</b> | 675M    | 100K      | 15.9        | 5.64        | 78.1         | 0.67        | 0.62        |
| <b>+HASTE</b> | 675M    | 200K      | 9.9         | 5.04        | 108.8        | 0.69        | 0.64        |
| <b>+HASTE</b> | 675M    | 250K      | 8.4         | 4.90        | 119.6        | 0.70        | <b>0.65</b> |
| <b>+HASTE</b> | 675M    | 400K      | 7.3         | 5.05        | 128.7        | 0.72        | 0.64        |
| <b>+HASTE</b> | 675M    | 500K      | <b>5.3</b>  | <b>4.72</b> | <b>148.5</b> | <b>0.73</b> | <b>0.65</b> |

Table 3: Additional evaluation results on ImageNet  $256 \times 256$ .  $\uparrow$  and  $\downarrow$  denote higher and lower values are better, respectively. **Bold font** denotes the best performance.

29 Additionally, we provide the results of SiT-XL/2+HASTE with different classifier-free guidance [4]  
 30 scales and intervals [9].

| model         | #params | iteration | interval  | CFG scale | FID↓        | sFID↓       | IS↑          | Prec.↑      | Rec.↑       |
|---------------|---------|-----------|-----------|-----------|-------------|-------------|--------------|-------------|-------------|
| SiT-XL/2      | 675M    | 7M        | [0, 1]    | 1.50      | 2.06        | 4.50        | 270.3        | <b>0.82</b> | 0.59        |
| <b>+HASTE</b> | 675M    | 500K      | [0, 1]    | 1.25      | 2.18        | 4.67        | 240.4        | 0.81        | 0.60        |
| <b>+HASTE</b> | 675M    | 500K      | [0, 0.7]  | 1.50      | 1.80        | 4.58        | 252.1        | 0.80        | 0.61        |
| <b>+HASTE</b> | 675M    | 500K      | [0, 0.6]  | 1.825     | 1.74        | 4.74        | 268.7        | 0.80        | 0.62        |
| <b>+HASTE</b> | 675M    | 2M        | [0, 0.7]  | 1.7       | 1.45        | 4.55        | 297.3        | 0.80        | 0.64        |
| <b>+HASTE</b> | 675M    | 2M        | [0, 0.7]  | 1.65      | 1.44        | 4.56        | 289.4        | 0.79        | 0.64        |
| <b>+HASTE</b> | 675M    | 2M        | [0, 0.7]  | 1.675     | 1.44        | 4.55        | 293.7        | 0.80        | 0.64        |
| <b>+HASTE</b> | 675M    | 2.5M      | [0, 0.7]  | 1.7       | 1.43        | 4.56        | 298.8        | 0.80        | 0.64        |
| <b>+HASTE</b> | 675M    | 2.5M      | [0, 0.7]  | 1.65      | 1.43        | 4.57        | 290.7        | 0.80        | 0.64        |
| <b>+HASTE</b> | 675M    | 2.5M      | [0, 0.72] | 1.65      | <b>1.42</b> | <b>4.49</b> | <b>299.5</b> | 0.80        | <b>0.65</b> |

Table 4: Evaluation results on ImageNet  $256 \times 256$  with different classifier-free guidance settings.

## 31 B Additional Implementation Details.

|                      | SiT-B                   | SiT-L                   | SiT-XL                  | DiT-XL                  |
|----------------------|-------------------------|-------------------------|-------------------------|-------------------------|
| <b>Architecture</b>  |                         |                         |                         |                         |
| input dim.           | $32 \times 32 \times 4$ | $32 \times 32 \times 4$ | $32 \times 32 \times 4$ | $32 \times 32 \times 4$ |
| num. layers          | 12                      | 24                      | 28                      | 28                      |
| hidden dim.          | 768                     | 1024                    | 1152                    | 1152                    |
| num. heads           | 12                      | 16                      | 16                      | 16                      |
| <b>HASTE</b>         |                         |                         |                         |                         |
| $\lambda_R$          | 0.5                     | 0.5                     | 0.5                     | 0.5                     |
| $\lambda_A$          | 0.5                     | 0.5                     | 0.5                     | 0.5                     |
| alignment depth      | 5                       | 8                       | 8                       | 8                       |
| student layers       | [2, 3, 4]               | [4, 5, 6, 7]            | [4, 5, 6, 7]            | [4, 5, 6, 7]            |
| teacher model        | DINOv2-B [14]           | DINOv2-B [14]           | DINOv2-B [14]           | DINOv2-B [14]           |
| teacher layers       | [7, 9, 11]              | [8, 9, 10, 11]          | [8, 9, 10, 11]          | [8, 9, 10, 11]          |
| termination iter.    | 100 K                   | 250 K                   | 250 K                   | 250 K                   |
| alignment heads      | 0-11                    | 0-11                    | 0-11                    | 0-11                    |
| <b>Optimization</b>  |                         |                         |                         |                         |
| batch size           | 256                     | 256                     | 256                     | 256                     |
| optimizer            | AdamW [7, 11]           | AdamW [7, 11]           | AdamW [7, 11]           | AdamW [7, 11]           |
| lr                   | 0.0001                  | 0.0001                  | 0.0001                  | 0.0001                  |
| $(\beta_1, \beta_2)$ | (0.9, 0.999)            | (0.9, 0.999)            | (0.9, 0.999)            | (0.9, 0.999)            |
| weight decay         | 0                       | 0                       | 0                       | 0                       |
| <b>Diffusion</b>     |                         |                         |                         |                         |
| objective            | linear interpolants     | linear interpolants     | linear interpolants     | improved DDPM           |
| prediction           | velocity                | velocity                | velocity                | noise and variance      |
| sampler              | Euler-Maruyama          | Euler-Maruyama          | Euler-Maruyama          | Euler-Maruyama          |
| sampling steps       | 250                     | 250                     | 250                     | 250                     |

Table 5: Detailed training settings.

32 **Further implementation details.** For XL and L-sized models, we set the feature alignment depth  
33 to 8 following REPA, and extract the attention maps from layer [4, 5, 6, 7] (counting from 0) of  
34 diffusion transformers, to align with those from layer [8, 9, 10, 11] of DINOv2-B. According to  
35 [10], the performance almost saturates when transferring 12 out of 16 heads, and the student can  
36 also develop its own attention patterns for unused heads. Specifically, since the number of heads  
37 for DINOv2-B layer is only 12, we conduct attention alignment partially over the first 12 heads of  
38 diffusion transformer layer. For B-sized models, the feature alignment depth is adjusted to 5, and we  
39 extract the attention maps from layer [2, 3, 4] to align with those from layer [7, 9, 11] of DINOv2-B.

40 We enable mixed-precision (fp16) for efficient training. For data pre-processing, we leverage the  
41 protocols provided in EDM2 [5] to pre-compute latent vectors from images with stable diffusion  
42 VAE [17]. Specifically, we use `stabilityai/sd-vae-ft-ema` decoder to translate generated latent  
43 vectors into images. Following REPA [19], we also use three-layer MLP with SiLU activations [2] as  
44 the projector of hidden states. For MM-DiT, we use CLIP [16] text model to encode captions.

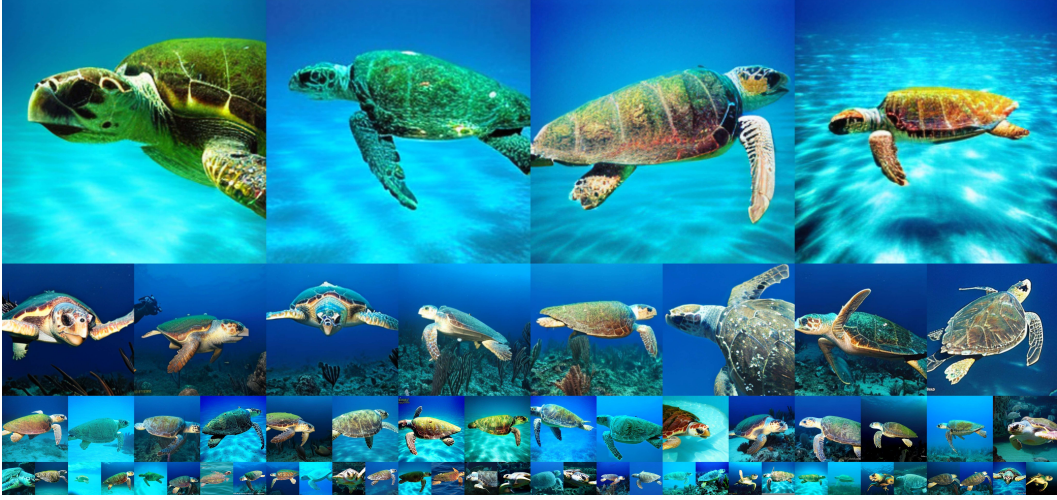


Figure 2: Uncurated generation results of SiT-XL/2+HASTE. We use classifier-free guidance with  $w = 4.0$ . Class label = “loggerhead sea turtle” (33).



Figure 3: Uncurated generation results of SiT-XL/2+HASTE. We use classifier-free guidance with  $w = 4.0$ . Class label = “macaw” (88).





Figure 4: Uncurated generation results of SiT-XL/2+HASTE. We use classifier-free guidance with  $w = 4.0$ . Class label = “golden retriever” (207).



Figure 5: Uncurated generation results of SiT-XL/2+HASTE. We use classifier-free guidance with  $w = 4.0$ . Class label = “arctic wolf” (270).



Figure 6: Uncurated generation results of SiT-XL/2+HASTE. We use classifier-free guidance with  $w = 4.0$ . Class label = “red panda” (387).



Figure 7: Uncurated generation results of SiT-XL/2+HASTE. We use classifier-free guidance with  $w = 4.0$ . Class label = “panda” (388).





Figure 8: Uncurated generation results of SiT-XL/2+HASTE. We use classifier-free guidance with  $w = 4.0$ . Class label = “acoustic guitar” (402).



Figure 9: Uncurated generation results of SiT-XL/2+HASTE. We use classifier-free guidance with  $w = 4.0$ . Class label = “balloon” (417).



Figure 10: Uncurated generation results of SiT-XL/2+HASTE. We use classifier-free guidance with  $w = 4.0$ . Class label = “baseball” (429).



Figure 11: Uncurated generation results of SiT-XL/2+HASTE. We use classifier-free guidance with  $w = 4.0$ . Class label = “dog sled” (537).





Figure 12: Uncurated generation results of SiT-XL/2+HASTE. We use classifier-free guidance with  $w = 4.0$ . Class label = “fire truck” (555).



Figure 13: Uncurated generation results of SiT-XL/2+HASTE. We use classifier-free guidance with  $w = 4.0$ . Class label = “laptop” (620).





Figure 14: Uncurated generation results of SiT-XL/2+HASTE. We use classifier-free guidance with  $w = 4.0$ . Class label = “space shuttle” (812).



Figure 15: Uncurated generation results of SiT-XL/2+HASTE. We use classifier-free guidance with  $w = 4.0$ . Class label = “cheeseburger” (933).



Figure 16: Uncurated generation results of SiT-XL/2+HASTE. We use classifier-free guidance with  $w = 4.0$ . Class label = “cliff drop-off” (972).

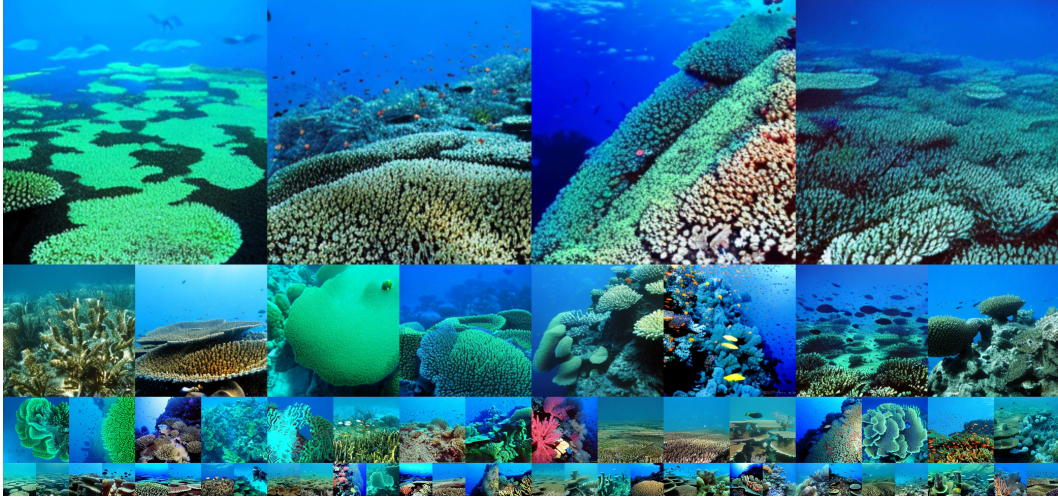


Figure 17: Uncurated generation results of SiT-XL/2+HASTE. We use classifier-free guidance with  $w = 4.0$ . Class label = “coral reef” (973).





Figure 18: Uncurated generation results of SiT-XL/2+HASTE. We use classifier-free guidance with  $w = 4.0$ . Class label = “lake shore” (975).



Figure 19: Uncurated generation results of SiT-XL/2+HASTE. We use classifier-free guidance with  $w = 4.0$ . Class label = “volcano” (980).

## References

- [1] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009.
- [2] Stefan Elfving, Eiji Uchibe, and Kenji Doya. Sigmoid-weighted linear units for neural network function approximation in reinforcement learning. *Neural Networks*, 107:3–11, 2018. ISSN 0893-6080. doi: <https://doi.org/10.1016/j.neunet.2017.12.012>. URL <https://www.sciencedirect.com/science/article/pii/S0893608017302976>. Special issue on deep reinforcement learning.
- [3] 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 *NeurIPS*, 2017.
- [4] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. In *NeurIPS Workshop*, 2021.
- [5] Tero Karras, Miika Aittala, Jaakko Lehtinen, Janne Hellsten, Timo Aila, and Samuli Laine. Analyzing and improving the training dynamics of diffusion models. In *CVPR*, 2024.
- [6] Dongjun Kim, Seungjae Shin, Kyungwoo Song, Wanmo Kang, and Il-Chul Moon. Soft truncation: A universal training technique of score-based diffusion model for high precision score estimation. In *ICML*, 2022.
- [7] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015.
- [8] Tuomas Kynkäänniemi, Tero Karras, Samuli Laine, Jaakko Lehtinen, and Timo Aila. Improved precision and recall metric for assessing generative models. In *NeurIPS*, 2019.
- [9] Tuomas Kynkäänniemi, Miika Aittala, Tero Karras, Samuli Laine, Timo Aila, and Jaakko Lehtinen. Applying guidance in a limited interval improves sample and distribution quality in diffusion models. In *NeurIPS*, 2024.
- [10] Alexander Cong Li, Yuandong Tian, Beidi Chen, Deepak Pathak, and Xinlei Chen. On the surprising effectiveness of attention transfer for vision transformers. In *NeurIPS*, 2024.
- [11] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *ICLR*, 2017.
- [12] 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. *arXiv preprint arXiv:2401.08740*, 2024. URL <https://arxiv.org/abs/2401.08740>.
- [13] Charlie Nash, Jacob Menick, Sander Dieleman, and Peter Battaglia. Generating images with sparse representations. In *ICML*, 2021.
- [14] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel HAZIZA, Francisco Massa, Alaaeldin El-Nouby, Mido Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Herve Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. DINOv2: Learning robust visual features without supervision. *TMLR*, 2024. ISSN 2835-8856. URL <https://openreview.net/forum?id=a68SUt6zFt>. Featured Certification.
- [15] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *ICCV*, 2023.
- [16] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *ICML*, 2021.
- [17] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, 2022.
- [18] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training gans. In *NeurIPS*, 2016.
- [19] 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 *ICLR*, 2025.