Supplementary Material for Adversarial Diffusion Distillation

Axel Sauer, Dominik Lorenz, Andreas Blattmann, and Robin Rombach

Stability AI

1 SDS As a Special Case of the Distillation Loss

If we set the weighting function to $c(t) = \frac{\alpha_t}{2\sigma_t} w(t)$ where $w(t)$ is the scaling factor from the weighted diffusion loss as in [14] and choose $d(x, y) = ||x - y||_2^2$, the distillation loss in Eq. 4 is equivalent to the score distillation objective:

$$
\begin{aligned}
&\frac{d}{d\theta}\mathcal{L}_{\text{distill}}^{\text{MSE}} \\
&\quad= \mathbb{E}_{t,\epsilon'}\Big[c(t)\frac{d}{d\theta}||\hat{x}_\theta - \hat{x}_\psi(\text{sg}(\hat{x}_{\theta,t}); t)||_2^2\Big] \\
&\quad= \mathbb{E}_{t,\epsilon'}\Big[2c(t)[\hat{x}_\theta - \hat{x}_\psi(\hat{x}_{\theta,t}; t)]\frac{d\hat{x}_\theta}{d\theta}\Big] \\
&\quad= \mathbb{E}_{t,\epsilon'}\Big[\frac{\alpha_t}{\sigma_t} w(t)[\frac{1}{\alpha_t}(\hat{x}_{\theta,t} - \hat{x}_{\theta,t}) + \hat{x}_\theta - \hat{x}_\psi(\hat{x}_{\theta,t}; t)]\frac{d\hat{x}_\theta}{d\theta}\Big] \\
&\quad= \mathbb{E}_{t,\epsilon'}\Big[\frac{1}{\sigma_t} w(t)[(\alpha_t\hat{x}_\theta - \hat{x}_{\theta,t}) - (\alpha_t\hat{x}_\psi(\hat{x}_{\theta,t}; t) - \hat{x}_{\theta,t})]\frac{d\hat{x}_\theta}{d\theta}\Big] \\
&\quad= \mathbb{E}_{t,\epsilon'}\Big[\frac{w(t)}{\sigma_t}[-\sigma_t\epsilon' + \sigma_t\hat{\epsilon}_\theta(\hat{x}_{\theta,t}; t)]\frac{d\hat{x}_\theta}{d\theta}\Big] \\
&\quad= \frac{d}{d\theta}\mathcal{L}_{\text{SDS}}
\end{aligned}
\tag{1}
$$

2 Details on Human Preference Assessment

For the evaluation results presented in Section 4, we employ human evaluation and do not rely on commonly used metrics for quality assessment of generative models such as FID [4] and CLIP-score [15], since these have been shown to capture more fine grained aspects like aesthetics and scene composition only insufficiently [7,13]. However these categories in particular have become more and more important when comparing current state-of-the-art text-to-image models. We evaluate all models based on 100 selected prompts from the PartiPrompts benchmark [18] with the most relevant categories (excluding prompts from the category *basic*). More details on how the study was conducted Section 2.1 and the rankings computed Section 2.2 are listed below.

Fig. 1: User preference study (*single step*). We compare the performance of ADD-M (1-step) against established baselines.

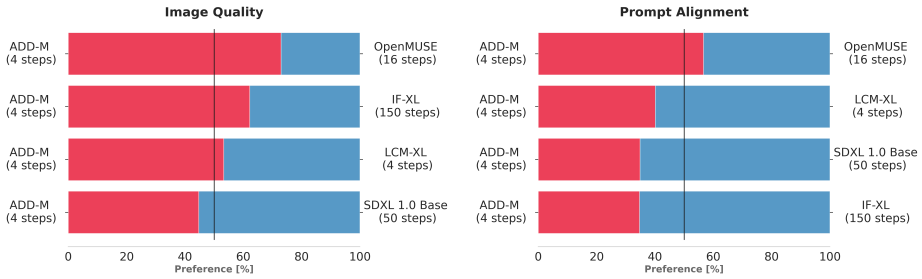

Fig. 2: User preference study (*multiple steps*). We compare the performance of ADD-XL (4-step) against established baselines.

2.1 Experimental Setup

Given all models for one particular study (e.g. ADD-XL, OpenMUSE[1], IF-XL[2], SDXL [13] and LCM-XL[3] [10,11] in Section 4.2) we compare each prompt for each pair of models (1v1). For every comparison, we collect an average of four votes per task from different annotators, for both visual quality and prompt following. Human evaluators, recruited from the platform *Prolific*[4] with English as their first language, are shown two images from different models based on the same text prompt. To prevent biases, evaluators are restricted from participating in more than one of our studies. For the prompt following task, we display the text prompt above the two images and ask, "Which image looks more representative of the text shown above and faithfully follows it?" For the visual quality assessment, we do not show the prompt and instead ask, "Which image is of higher quality and aesthetically more pleasing?". Performing a complete assessment between all pair-wise comparisons gives us robust and reliable signals on model performance trends and the effect of varying thresholds. The order of prompts and the order

[1] https://huggingface.co/openMUSE
[2] https://github.com/deep-floyd/IF
[3] https://huggingface.co/latent-consistency/lcm-lora-sdxl
[4] https://app.prolific.com

between models are fully randomized. Frequent attention checks are in place to ensure data quality.

2.2 ELO Score Calculation

To calculate rankings when comparing more than two models based on 1v1 comparisons we use ELO Scores (higher-is-better) [3] which were originally proposed as a scoring method for chess players but have more recently also been applied to compare instruction-tuned generative LLMs [1, 2]. For a set of competing players with initial ratings R_{init} participating in a series of zero-sum games the ELO rating system updates the ratings of the two players involved in a particular game based on the expected and and actual outcome of that game. Before the game with two players with ratings R_1 and R_2, the expected outcome for the two players are calculated as

$$E_1 = \frac{1}{1 + 10^{\frac{R_2 - R_1}{400}}}\,, \tag{2}$$

$$E_2 = \frac{1}{1 + 10^{\frac{R_1 - R_2}{400}}}\,. \tag{3}$$

After observing the result of the game, the ratings R_i are updated via the rule

$$R_i^{'} = R_i + K \cdot (S_i - E_i)\,, \quad i \in \{1, 2\} \tag{4}$$

where S_i indicates the outcome of the match for player i. In our case we have $S_i = 1$ if player i wins and $S_i = 0$ if player i looses. The constant K can be see as weight putting emphasis on more recent games. We choose $K = 1$ and bootstrap the final ELO ranking for a given series of comparisons based on 1000 individual ELO ranking calculations with randomly shuffled order. Before comparing the models we choose the start rating for every model as $R_{\text{init}} = 1000$.

3 GAN Baselines Comparison

For training our state-of-the-art GAN baseline StyleGAN-T++, we follow the training procedure outlined in [16]. The main differences are extended training (∼2M iterations with a batch size of 2048, which is comparable to GigaGAN's schedule [5]), the improved discriminator architecture proposed in Section 3, and R1 penalty applied at each discriminator head.

Fig. 3 shows that StyleGAN-T++ outperforms the previous best GANs by achieving a comparable zero-shot FID to GigaGAN at a significantly higher CLIP score. Here, we do not compare to DMs, as comparisons between model classes via automatic metrics tend to be less informative [17]. As an example, GigaGAN achieves FID and CLIP scores comparable to SD1.5, but its sample quality is still inferior, as noted by the authors.

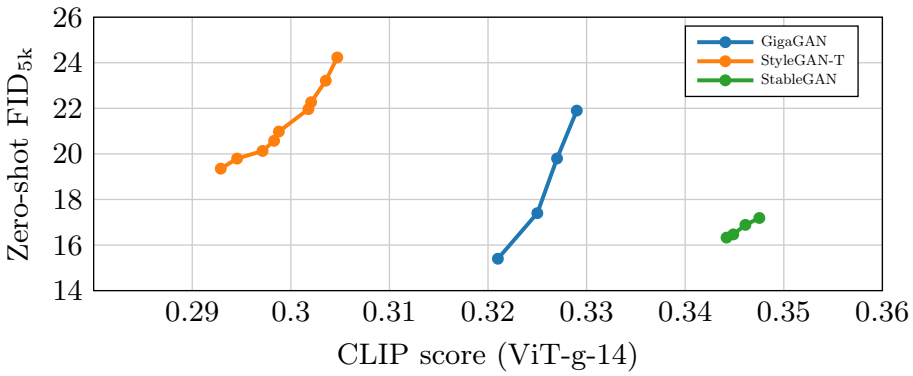

Fig. 3: Comparing text alignment tradeoffs at 256 × 256 pixels. We compare FID–CLIP score curves of StyleGAN-T, StyleGAN-T++, and GigaGAN. For increasing CLIP score, all methods use via decreasing truncation [6] for values $\psi = \{1.0, 0.9, \ldots, 0.3\}$.

Fig. 4: Additional single step 512^2 images generated with ADD-XL. All samples are generated with a single U-Net evaluation trained with adversarial diffusion distillation (ADD).

4 Additional Samples

We show additional one-step samples in Figure 4. An additional qualitative comparison which demonstrates that our model can further refine quality by using more than one sampling step is provided in Figure 6, where we show that, while sampling quality with a single step is already high, more steps can give higher diversity and better spelling capabilities. Lastly, we provide an additional qualitative comparison of ADD-XL to other state-of-the-art one and few-step models in Figure 5.

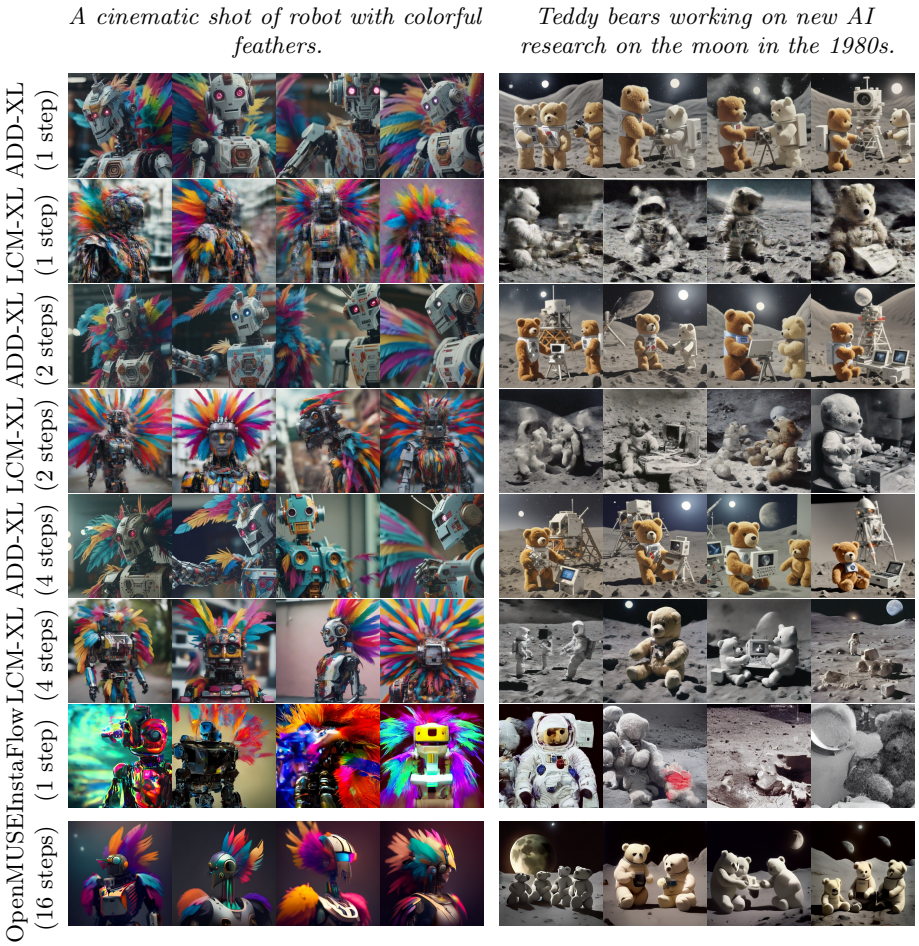

Fig. 5: Additional qualitative comparisons to state of the art fast samplers. Few step samples from our ADD-XL and LCM-XL [11], InstaFlow [9], and OpenMuse [12].

Fig. 6: Additional results on the qualitative effect of sampling steps. We show qualitative examples when sampling ADD-XL with 1, 2, and 4 steps. Single-step samples are often already of high quality, but increasing the number of steps can further improve the diversity (left) and spelling capabilities (right). The seeds are constant within columns and we see that the general layout is preserved across sampling steps, allowing for fast exploration of outputs while retaining the possibility to refine.

5 Additional implementation details

- The final model *ADD-XL* is trained for 2 days on 16 A100s 80GB and a batch size of 2 per GPU.
- The discriminator heads are applied after each network block of the DINO ViT, ie., right after each residual MLP.
- The distillation weight is fairly tolerant; values between 1 and 50 are effective. We find that values below this range do not significantly improve text alignment, while larger values tend to introduce artifacts.
- We chose $N = 4$ because many recent works on few-shot synthesis (e.g. LCM) are trained in this range.
- The student model uses EMA.
- We used exponential weighting in Table 1 (d), but we do not obseve substantial differences for other weightings when using only $\mathcal{L}_{dist}$.

6 Quantitative evaluation of sample diversity.

We assess sample diversity via recall on COCO2014 [8]. We plot sampling steps over recall with the teacher at 50 steps as the upper bound in Fig. 7. These additional results underline our claims that diversity is reduced compared to the teacher model underlining our qualitative comparisons.

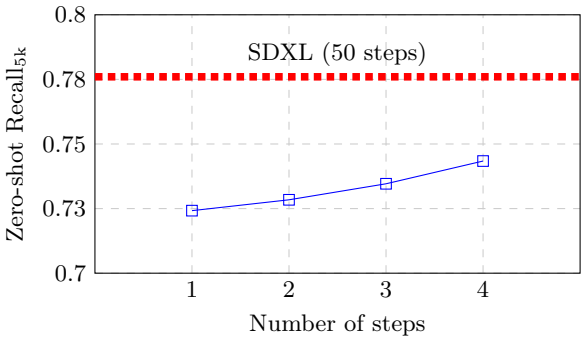

Fig. 7: Zero-shot recall over steps evaluated on COCO.

7 Training algorithm.

Algorithm 1 Training Procedure

Require: Pretrained DM weights θ, trainable discriminator weights ϕ, frozen DM teacher weights ψ
Require: Dataset of real images x_0, diffusion coefficients α_s, σ_s, and timesteps T_{student}
Ensure: Optimized student model weights and discriminator
Initialize ADD-student with weights θ
Initialize discriminator with weights ϕ
Freeze DM teacher weights ψ
for each training step **do**
    Sample s uniformly from T_{student}
    Produce noisy data $x_s = \alpha_s x_0 + \sigma_s \epsilon$
    Generate samples $\hat{x}_\theta(x_s, s)$ using ADD-student
    Pass $\hat{x}_\theta$ and x_0 to discriminator
    Compute adversarial loss $\mathcal{L}^{\text{G}}_{\text{adv}}$
    Diffuse $\hat{x}_\theta$ with teacher's process to $\hat{x}_{\theta,t}$
    Compute distillation loss $\mathcal{L}_{\text{distill}}$ using teacher's prediction
    Update model weights by minimizing $\mathcal{L} = \mathcal{L}^{\text{G}}_{\text{adv}}(\hat{x}_\theta(x_s, s), \phi) + \lambda \mathcal{L}_{\text{distill}}(\hat{x}_\theta(x_s, s), \psi)$
end for

References

1. Askell, A., Bai, Y., Chen, A., Drain, D., Ganguli, D., Henighan, T., Jones, A., Joseph, N., Mann, B., DasSarma, N., Elhage, N., Hatfield-Dodds, Z., Hernandez, D., Kernion, J., Ndousse, K., Olsson, C., Amodei, D., Brown, T., Clark, J., McCandlish, S., Olah, C., Kaplan, J.: A general language assistant as a laboratory for alignment (2021)

2. Bai, Y., Jones, A., Ndousse, K., Askell, A., Chen, A., DasSarma, N., Drain, D., Fort, S., Ganguli, D., Henighan, T., Joseph, N., Kadavath, S., Kernion, J., Conerly, T., El-Showk, S., Elhage, N., Hatfield-Dodds, Z., Hernandez, D., Hume, T., Johnston, S., Kravec, S., Lovitt, L., Nanda, N., Olsson, C., Amodei, D., Brown, T., Clark, J., McCandlish, S., Olah, C., Mann, B., Kaplan, J.: Training a helpful and harmless assistant with reinforcement learning from human feedback (2022)
3. Elo, A.E.: The Rating of Chessplayers, Past and Present. Arco Pub., New York (1978)
4. Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., Hochreiter, S.: GANs trained by a two time-scale update rule converge to a local Nash equilibrium. NeurIPS (2017)
5. Kang, M., Zhu, J.Y., Zhang, R., Park, J., Shechtman, E., Paris, S., Park, T.: Scaling up gans for text-to-image synthesis. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 10124–10134 (2023)
6. Karras, T., Laine, S., Aila, T.: A style-based generator architecture for generative adversarial networks. 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) pp. 4396–4405 (2018), https://api.semanticscholar.org/CorpusID:54482423
7. Kirstain, Y., Polyak, A., Singer, U., Matiana, S., Penna, J., Levy, O.: Pick-a-pic: An open dataset of user preferences for text-to-image generation (2023)
8. Kynkäänniemi, T., Karras, T., Laine, S., Lehtinen, J., Aila, T.: Improved precision and recall metric for assessing generative models. Advances in neural information processing systems **32** (2019)
9. Liu, X., Zhang, X., Ma, J., Peng, J., Liu, Q.: Instaflow: One step is enough for high-quality diffusion-based text-to-image generation. arXiv preprint arXiv:2309.06380 (2023)
10. Luo, S., Tan, Y., Huang, L., Li, J., Zhao, H.: Latent consistency models: Synthesizing high-resolution images with few-step inference. ArXiv **abs/2310.04378** (2023), https://api.semanticscholar.org/CorpusID:263831037
11. Luo, S., Tan, Y., Patil, S., Gu, D., von Platen, P., Passos, A., Huang, L., Li, J., Zhao, H.: Lcm-lora: A universal stable-diffusion acceleration module. ArXiv **abs/2311.05556** (2023), https://api.semanticscholar.org/CorpusID:265067414
12. Patil, S., Berman, W., von Platen, P.: Amused: An open muse model. https://github.com/huggingface/diffusers (2023)
13. Podell, D., English, Z., Lacey, K., Blattmann, A., Dockhorn, T., Müller, J., Penna, J., Rombach, R.: Sdxl: Improving latent diffusion models for high-resolution image synthesis. arXiv preprint arXiv:2307.01952 (2023)
14. Poole, B., Jain, A., Barron, J.T., Mildenhall, B.: Dreamfusion: Text-to-3d using 2d diffusion. arXiv preprint arXiv:2209.14988 (2022)
15. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al.: Learning transferable visual models from natural language supervision. In: International conference on machine learning. pp. 8748–8763. PMLR (2021)
16. Sauer, A., Karras, T., Laine, S., Geiger, A., Aila, T.: Stylegan-t: Unlocking the power of gans for fast large-scale text-to-image synthesis. Proc. ICML (2023)
17. Stein, G., Cresswell, J.C., Hosseinzadeh, R., Sui, Y., Ross, B.L., Villecroze, V., Liu, Z., Caterini, A.L., Taylor, J.E.T., Loaiza-Ganem, G.: Exposing flaws of generative model evaluation metrics and their unfair treatment of diffusion models. arXiv preprint arXiv:2306.04675 (2023)

18. Yu, J., Xu, Y., Koh, J.Y., Luong, T., Baid, G., Wang, Z., Vasudevan, V., Ku, A., Yang, Y., Ayan, B.K., Hutchinson, B., Han, W., Parekh, Z., Li, X., Zhang, H., Baldridge, J., Wu, Y.: Scaling autoregressive models for content-rich text-to-image generation (2022)