TurboFill: Adapting Few-step Text-to-image Model for Fast Image Inpainting

Supplementary Material

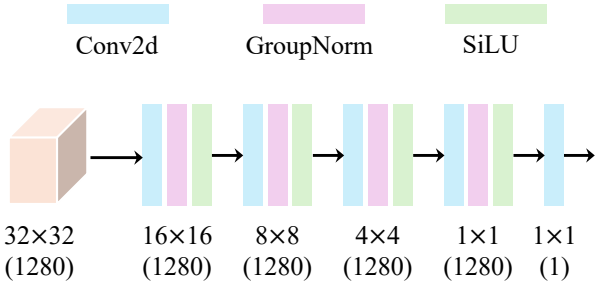

Figure 1. Network structure of classifier.

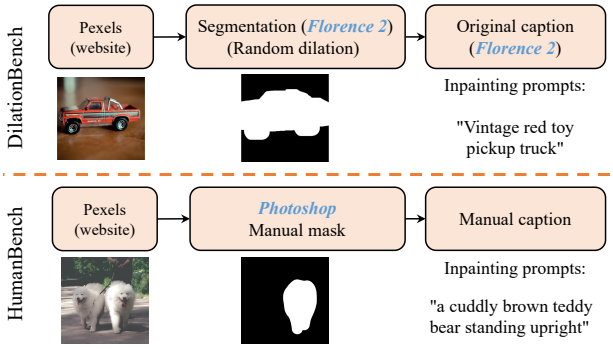

Figure 2. Data collection process.

In this supplementary file, we provide the following materials:

1. Details of classifier.
2. Descriptions of DilationBench and HumanBench.
3. Comparison with lora-based few-step image inpainting models.
4. Discussion of FID and user study.
5. More qualitative comparisons on DilationBench and HumanBench.
6. More visual results for ablation studies.

1. Details of Classifier

As mentioned in the main paper, TurboFill integrates both GAN and diffusion losses to enhance image realism and style coherence. To compute the GAN loss, we map the feature output from the assistant encoder into a one-dimensional vector using a classifier, shown in Fig. 1. Specifically, the classifier consists of Conv2d layers, GroupNorm layers [12], and SiLU activation layers [5]. Starting with the feature map from the assistant encoder, the classifier uses 5 Conv2d layers to progressively reduce the spatial dimensions from from 32×32 to 1×1.

2. Descriptions of DilationBench and HumanBench

As shown in Fig. 2, the images of DilationBench and HumanBench are crawled from Pexels[1] .

For dilationBench, we employ Florence2 [13] to perform the dense region caption task using the prompt <DENSE_REGION_CAPTION>. This task localizes primary objects in the images and generates concise textual descriptions for them. These descriptions are subsequently fed into SAM [9] to extract the corresponding segmentation masks. Based on the obtained masks, we first perform the erosion operation using an 8×8 kernel of ones (all-ones kernel), followed by a dilation operation to generate the final segmentation mask. Each operation is applied for 2 iterations. The inpainting prompt describes the content within the segmentation region of the original image.

For humanbench, we adopt the photoshop and manually label the segmentation mask. The inpainting prompt is also manually written.

3. Comparison with LoRA-based Few-Step Image Inpainting Models

For inpainting methods [1, 4] that train the base model, directly replacing the base model with few-step diffusion models leads to poor results. However, these models can benefit from acceleration by using few-step LoRA [6]. Specifically, for BLD and SDXL-Inpainting, we utilize the 4-step LoRA released by DMD2 [14], combined with the LCMScheduler [8], to construct 4-step versions of BLD [1] and SDXL-Inpainting [4]. PowerPaint V2 is based on SD 1.5 [11] and BrushNet. Since DMD2 only offers a LoRA version compatible with SDXL, we use the acceleration LoRA provided by HyperSD [10]. The quantitative results for these three LoRA-based few-step inpainting models are presented in the Tab. 1 and Tab. 2.

The results on DilationBench and HumanBench reveal that the 4-step models accelerated using LoRA exhibit a significant performance drop compared to the original 50-step models, with the 4-step PowerPaint V2 performing the worst. Moreover, the acceleration achieved with LoRA falls far short of that achieved by models capable of replacing the base model, such as BrushNet. Among these few-step image inpainting models, TurboFill demonstrates the best performance.

† Corresponding author

[1] https://www.pexels.com/

Metrics		Mask Region Quality			Whole Image Quality			Text-Align
Method		Q-Align	CLIPIQA+	TOPIQ	Q-Align	CLIPIQA	TOPIQ	CLIP Sim
Multi-Step (50 steps)	BLD	4.1836	0.6874	5.1781	4.3320	0.6791	5.4662	24.884
	HD-Painter	3.9727	0.6611	4.9345	4.4805	0.6439	5.4566	25.585
	SDXL-Inpainting	4.2461	0.6666	5.1533	4.6172	0.6695	5.5409	24.848
	BrushNet-Rand	4.2617	0.6729	5.1738	4.6289	0.6818	5.5779	21.634
	BrushNet*	4.4492	0.7139	5.2744	4.6211	0.6958	5.5658	25.389
	PowerPaint V2	**4.7773**	**0.7765**	**5.5468**	**4.7227**	**0.7371**	**5.6474**	**26.256**
Few-Step (4 steps)	BLD	3.5469	0.5552	4.9605	4.0312	0.6321	5.4149	24.677
	SDXL-Inpainting	3.5469	0.5024	5.0080	4.1289	0.5930	5.4034	24.726
	PowerPaint V2	2.7949	0.5958	4.8472	3.3164	0.6366	5.1755	22.279
	BrushNet-Rand	4.1602	0.6538	4.9927	4.5547	0.6608	5.4599	21.831
	BrushNet*	4.1836	0.6572	4.9939	4.4492	0.6427	5.3961	25.341
	Ours	**4.5703**	**0.7332**	**5.2753**	**4.7188**	**0.7111**	**5.5392**	**25.352**

Table 1. Quantitative comparisons among TurboFill and other diffusion based inpainting models in DilationBench. **Red** and **blue** indicates the best multi-step and the best few-step performances, respectively.

Metrics		Mask Region Quality			Whole Image Quality			Text-Align
Method		Q-Align	CLIPIQA+	TOPIQ	Q-Align	CLIPIQA	TOPIQ	CLIP Sim
Multi-Step (50 steps)	BLD	4.1523	0.6908	5.2996	3.9062	0.6579	5.4199	24.767
	HD-Painter	3.9844	0.6493	5.0873	4.2070	0.6119	5.4381	25.996
	SDXL-Inpainting	3.9551	0.6405	5.2040	4.0469	0.6288	5.4565	24.041
	BrushNet*	4.2578	0.7063	5.4340	4.0898	0.6694	5.5480	25.366
	PowerPaint V2	**4.5586**	**0.7529**	**5.5475**	**4.3633**	**0.7112**	**5.6338**	**26.264**
Few-Step (4 steps)	BLD	3.8438	0.5963	5.1622	3.8242	0.6139	5.3520	25.244
	SDXL-Inpainting	3.1816	0.4363	4.9627	3.4805	0.5113	5.2137	24.057
	PowerPaint V2	2.7012	0.5641	4.7546	2.9180	0.5855	4.9632	20.847
	BrushNet*	4.0508	0.6327	5.1041	4.1484	0.6003	5.3505	25.473
	Ours	**4.4727**	**0.7257**	**5.3865**	**4.3203**	**0.6822**	**5.4992**	**25.710**

Table 2. Quantitative comparisons among TurboFill and other diffusion based inpainting models in HumanBench. **Red** and **blue** indicates the best multi-step and the best few-step performances, respectively.

We also visualize the comparison of few-step image inpainting methods in Fig. 3. It is evident that SDXL-Inpainting and PowerPaint V2 produce results with poor details and often fail to align with the prompt (e.g., rows 2 and 4). The results of BLD are slightly better, but they still exhibit noticeable artifacts (e.g., rows 2, 3 and 5) and occasionally generate outputs completely misaligned with the prompt (e.g., row 6). Similarly, BrushNet* sometimes aligns only partially with the prompt (e.g., rows 1 and 3). In contrast, TurboFill consistently produces prompt-aligned results with realistic details, rich textures, and seamless scene harmonization.

4. Discussion of FID and User Study

Fréchet Inception Distance (FID), which measures the distance between feature distributions of generated images and a ground truth (GT) dataset, is adopted as a primary evaluation metric in many Image Inpainting works [2, 3]. However, we find that FID does not reliably reflect the visual quality of results.

To investigate this issue, based on 5 images (Figure 3 in the main paper), we calculate FID scores of four methods. The results as shown in Tab. 3. Our analysis shows that when we use the original images (5 images) as GT, BrushNet* (50 steps) achieves the best performance while TurboFill is visually better. However, when we switch to a different GT dataset (300 images within DilationBench), TurboFill performs the best. This indicates that FID is highly sensitive to the choice of GT and, therefore, is not a reliable metric for evaluating inpainting results.

Considering that the ultimate goal of existing metrics is to align with human rater preferences, we directly conduct the user study to evaluate the results of different image inpainting methods. Specifically, we design two separate user studies: one comparing TurboFill with multi-step methods and the other comparing TurboFill with few-step methods. For each comparison group in the user study, we randomly shuffle the order of results from all methods and ask participants to select the highest-quality and most natural image, aligned with the prompt. Each user study includes 30 groups of images, and we invite 20 participants to take part in the evaluation.

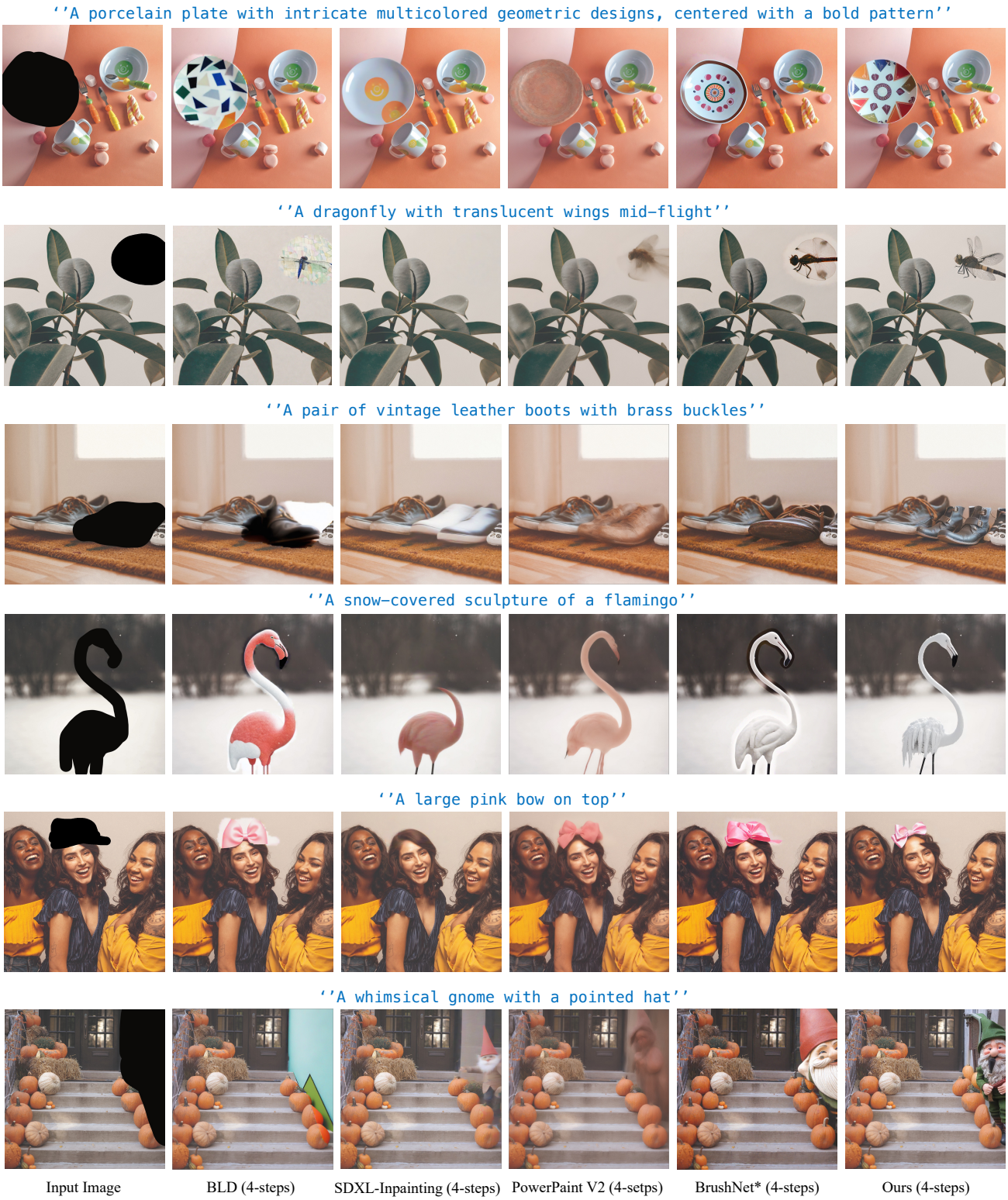

Figure 3. Comparison of few-step image inpainting methods on DilationBench. Compared to other few-step image inpainting models, TurboFill produces results that align more effectively with the prompt. Furthermore, TurboFill generates more realistic details and textures while achieving effective scene harmonization. (**Zoom in for best view**)

Methods	Multi-step (50 steps)		Few-step (4 steps)	
Metrics	BrushNet*	PowerPaint V2	BrushNet*	TurboFill
FID ↓ (5 images)	**29.3350**	37.4552	31.8095	33.1541
FID ↓ (300 images)	90.6107	91.9412	89.8712	**89.8300**

Table 3. Evaluation results based on the FID metric. The FID scores exhibit significant variability when applied to different GT datasets, indicating that FID is not a suitable metric for assessing diffusion-based image inpainting tasks.

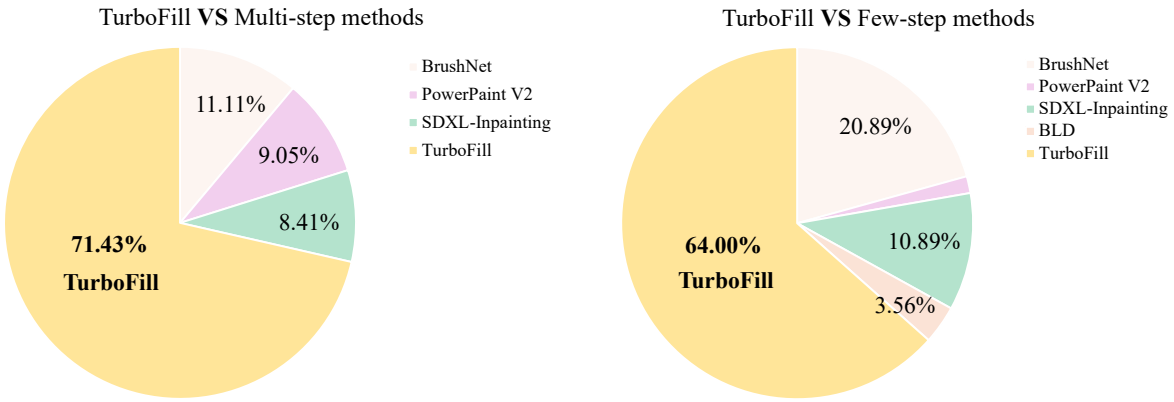

Figure 4. The results of user studies. We design two separate user studies: one comparing TurboFill with multi-step methods (left pie chart) and the other comparing TurboFill with few-step methods (right pie chart). It is evident that TurboFill's results are more favored by participants.

The results of user study are shown in Fig. 4. When comparing against multi-step image inpainting methods, including BrushNet [7], PowerPaint V2 [2], and SDXL-Inpainting [4], over 70% of participants prefer TurboFill, as shown in the left pie chart. They highlight its ability to produce more natural and detail-rich results. When comparing against few-step methods, including BrushNet, PowerPaint V2, SDXL-Inpainting, and BLD [1], TurboFill is favored by approximately 64% of participants, as shown in the right pie chart. Notably, PowerPaint V2, when accelerated with HyperSD's LoRA [10], generates blurred results with a lack of high-frequency details, as seen in Fig. 3, which likely contributes to its lower preference.

5. More Qualitative Comparisons

We present additional qualitative comparisons based on DilationBench (Fig. 5) and HumanBench (Fig. 6, Fig. 7). For DilationBench, SDXL-Inpainting and BrushNet* (50 steps) often only partially reflect the prompt content in their results (e.g., rows 4 and 6). PowerPaint V2 exhibits significant distortion issues (e.g., rows 1, 3, and 5), while BrushNet* (4 steps) occasionally produces oversaturated results (e.g., row 3). In contrast, our method demonstrates excellent detail preservation (e.g., the fur of animals) and achieves a harmonious overall image without overexposure.

For DilationBench, as shown in Fig. 6 and Fig. 7, it is observed that SDXL-Inpainting often fills the background into the inpainted areas (e.g., rows 3, 4, and 5 (Fig. 6)). In comparison, BrushNet* (50 steps) frequently produces results misaligned with the prompt (e.g., rows 3 and 5 (Fig. 6), row 3 (Fig. 7)) and introduces noticeable artifacts (e.g., row 2 (Fig. 6)). PowerPaint V2, on the other hand, generates results with significant distortions (e.g., rows 1 and 3 (Fig. 6), row 5 (Fig. 7)). BrushNet* (4 steps) exhibits evident overexposure issues (e.g., rows 1, 2, and 4 (Fig. 6), row 5 (Fig. 7)). Unlike other methods, TurboFill produces results that are more harmonious, with richer details, and effectively adheres to the prompt.

6. Visual Results for Ablation Studies

Starting from TurboFill, we remove $\mathcal{L}_{\text{BG}}$, $\mathcal{L}^F_{\text{Diff}}$, and $\mathcal{L}^R_{\text{Diff}}$ in sequence, with qualitative results shown in Fig. 8. In the visualizations, we see that without $\mathcal{L}_{\text{BG}}$, the color of background region changes noticeably, creating a sharp boundary between the fill-in and background regions. Further removing $\mathcal{L}^F_{\text{Diff}}$ introduces conflicting elements (i.e., house) in the fill-in region, suggesting the discriminator fails to fully capture the holistic scene. Finally, without $\mathcal{L}^R_{\text{Diff}}$, relying only on GAN loss, the inpainted images exhibit not only inconsistencies with the background but also poor texture and detail. This highlights that GAN loss alone struggles to close the gap between fake and real latents. Only when combining $\mathcal{L}^F_{\text{Diff}}$, $\mathcal{L}^R_{\text{Diff}}$, and $\mathcal{L}_{\text{BG}}$ in GAN training does the model achieve enhanced texture, detail, and effective scene harmonization between fill-in and background regions.

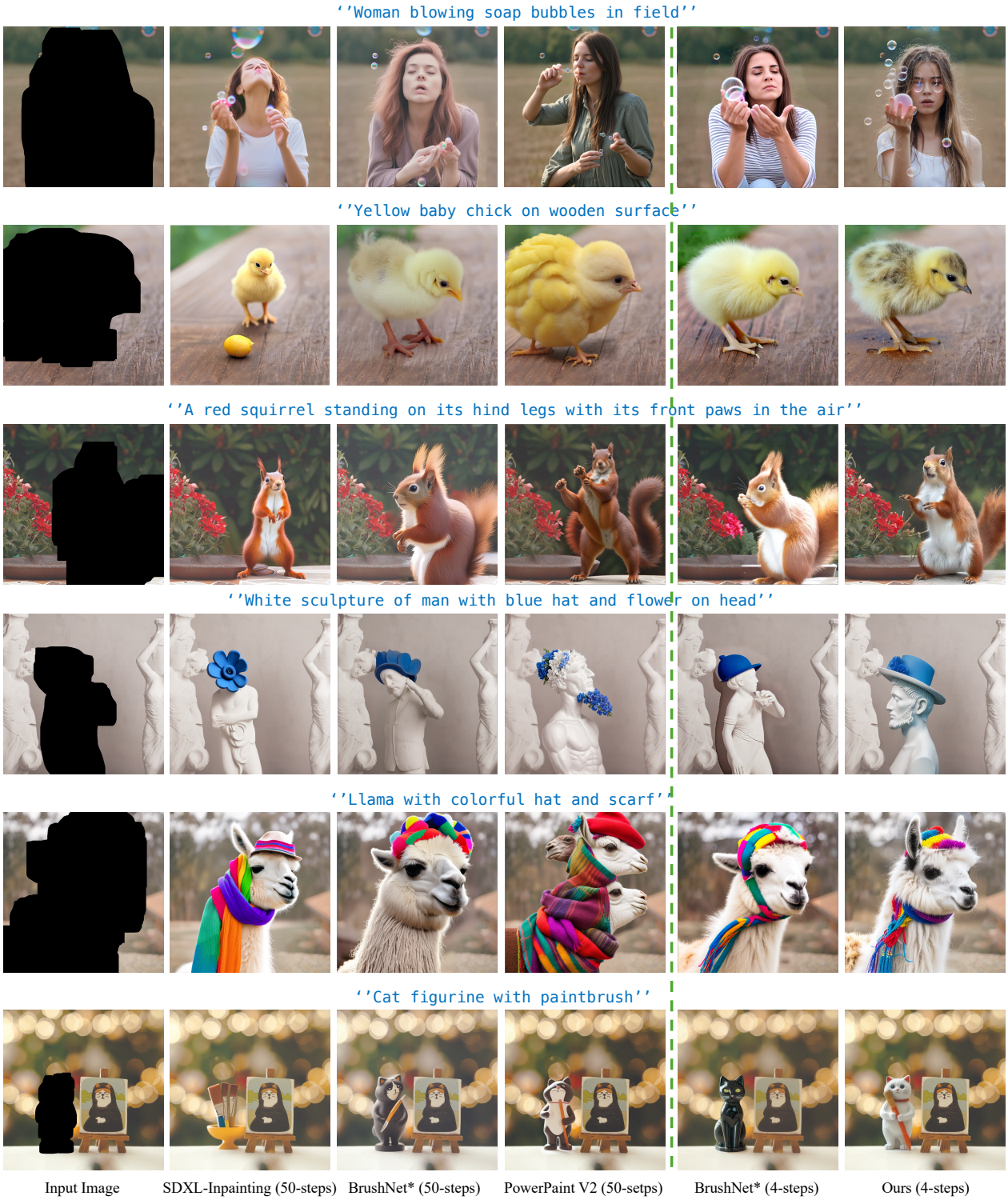

Figure 5. Comparison of previous inpainting methods and BrushNet on DilationBench. Compared to other methods, TurboFill generates more realistic details and textures in just 4 steps, while achieving good scene harmonization. (**Zoom in for best view**)

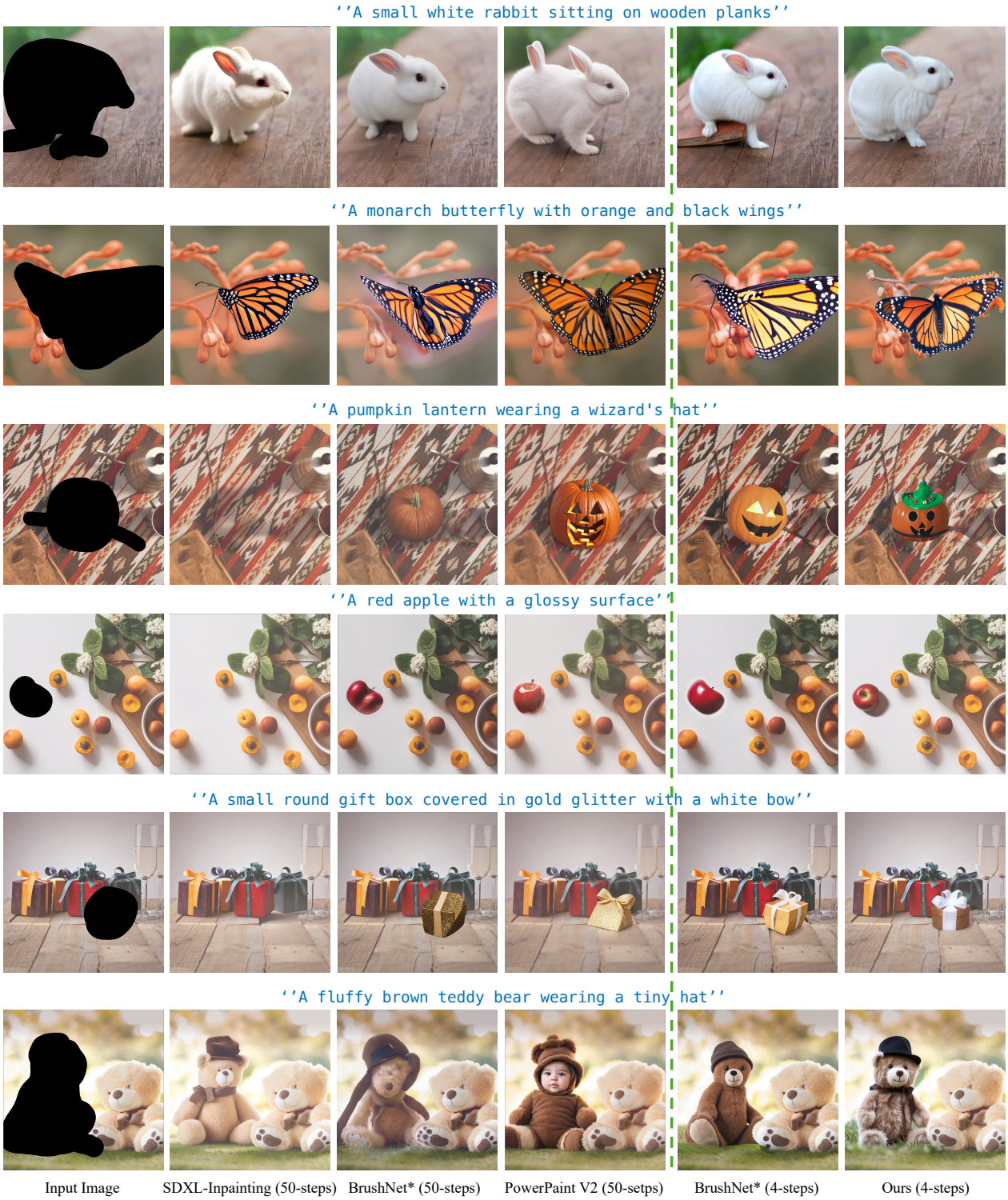

Figure 6. Comparison of previous inpainting methods and BrushNet on HumanBench. Compared to other methods, TurboFill generates more realistic details and textures in just 4 steps, while achieving good scene harmonization. (**Zoom in for best view**)

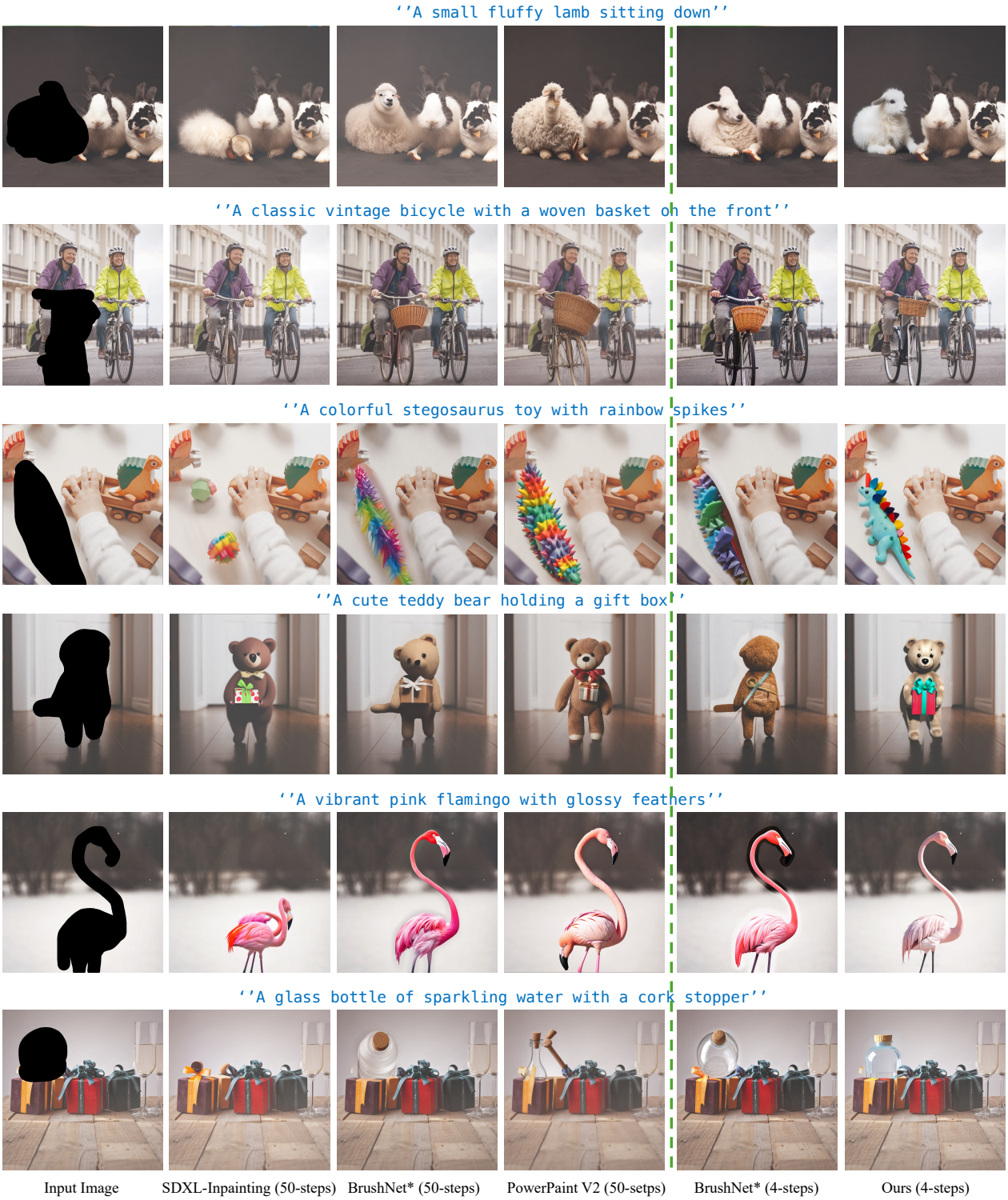

Figure 7. Comparison of previous inpainting methods and BrushNet on HumanBench. Compared to other methods, TurboFill generates more realistic details and textures in just 4 steps, while achieving good scene harmonization. (**Zoom in for best view**)

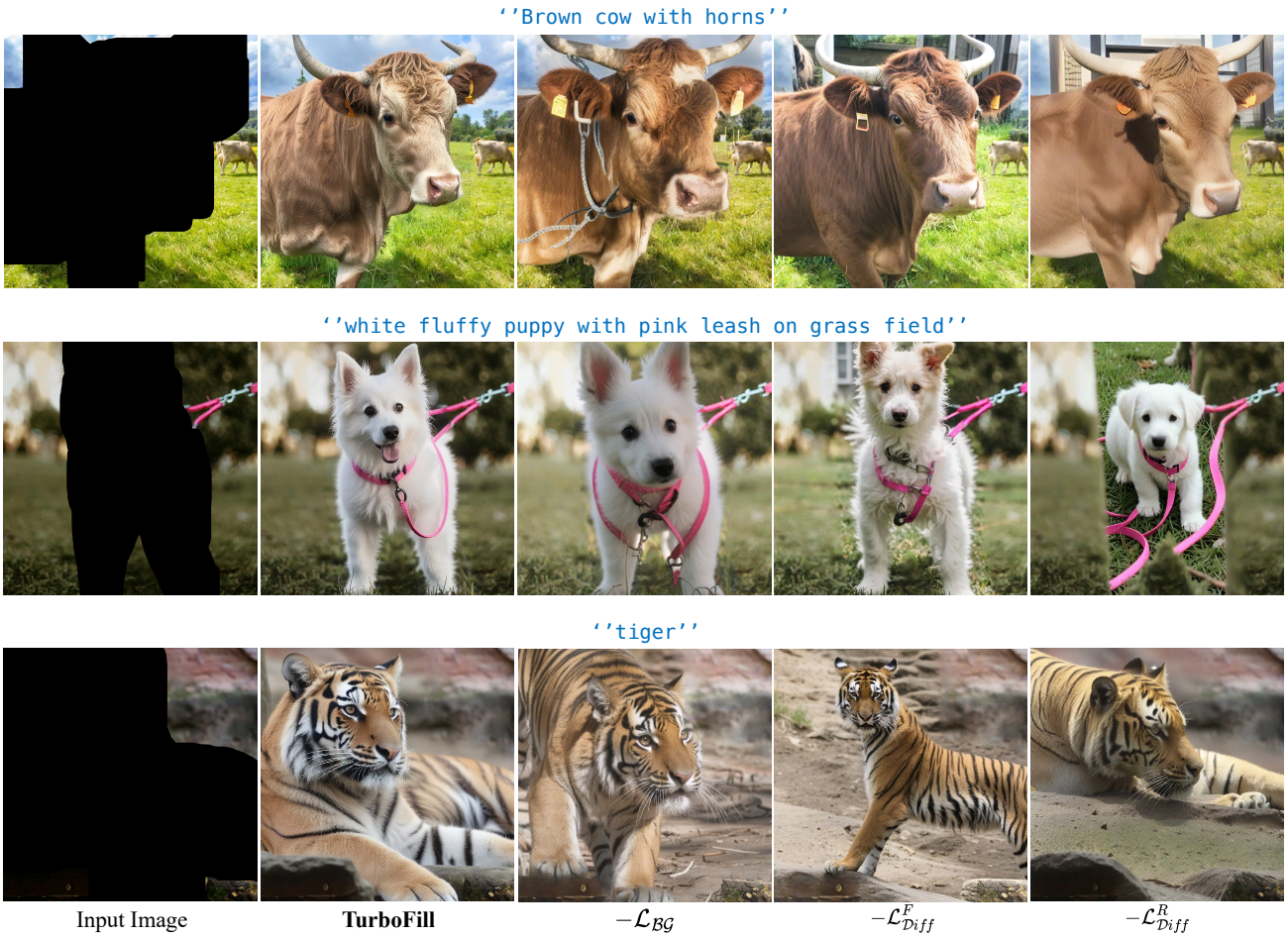

Figure 8. The effectiveness of different losses. From left to right, we progressively remove specific losses. (**Zoom in for best view**)

References

[1] Omri Avrahami, Ohad Fried, and Dani Lischinski. Blended latent diffusion. *ACM transactions on graphics (TOG)*, 42 (4):1–11, 2023. 1, 4

[2] Li Chen, Yu Wang, Jiali Zhang, and Xingxing Xu. Powerpaint: Adaptive task prompts for generalizable image inpainting. *arXiv preprint arXiv:2401.08965*, 2024. 2, 4

[3] Yifu Chen, Jingwen Chen, Yingwei Pan, Yehao Li, Ting Yao, Zhineng Chen, and Tao Mei. Improving text-guided object inpainting with semantic pre-inpainting. In *European Conference on Computer Vision*, pages 110–126. Springer, 2025. 2

[4] Hugggingface Diffusers. Sdxl-inpaint. `https://huggingface.co/diffusers/stable-diffusion-xl-1.0-inpainting-0.1`, Year. Accessed: YYYY-MM-DD. 1, 4

[5] Stefan Elfwing, Eiji Uchibe, and Kenji Doya. Sigmoid-weighted linear units for neural network function approximation in reinforcement learning. *Neural networks*, 107:3–11, 2018. 1

[6] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021. 1

[7] Xuan Ju, Xian Liu, Xintao Wang, Yuxuan Bian, Ying Shan, and Qiang Xu. Brushnet: A plug-and-play image inpainting model with decomposed dual-branch diffusion. *arXiv preprint arXiv:2403.06976*, 2024. 4

[8] Simian Luo, Yiqin Tan, Longbo Huang, Jian Li, and Hang Zhao. Latent consistency models: Synthesizing high-resolution images with few-step inference. *arXiv preprint arXiv:2310.04378*, 2023. 1

[9] Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, et al. Sam 2: Segment anything in images and videos. *arXiv preprint arXiv:2408.00714*, 2024. 1

[10] Yuxi Ren, Xin Xia, Yanzuo Lu, Jiacheng Zhang, Jie Wu, Pan Xie, Xing Wang, and Xuefeng Xiao. Hyper-sd: Trajectory segmented consistency model for efficient image synthesis. *arXiv preprint arXiv:2404.13686*, 2024. 1, 4

[11] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022. 1

[12] Yuxin Wu and Kaiming He. Group normalization. In *Proceedings of the European conference on computer vision (ECCV)*, pages 3–19, 2018. 1

[13] Bin Xiao, Haiping Wu, Weijian Xu, Xiyang Dai, Houdong Hu, Yumao Lu, Michael Zeng, Ce Liu, and Lu Yuan. Florence-2: Advancing a unified representation for a variety of vision tasks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4818–4829, 2024. 1

[14] Tianwei Yin, Michaël Gharbi, Taesung Park, Richard Zhang, Eli Shechtman, Fredo Durand, and William T Freeman. Improved distribution matching distillation for fast image synthesis. *arXiv preprint arXiv:2405.14867*, 2024. 1