Zero-shot Generative Model Adaptation via Image-specific Prompt Learning Supplementary Materials

A. Detailed Experiment Settings

We introduce the implementation details of GAN-IPL, Diff-IPL, and the evaluation metrics.

GAN-IPL. Our method is developed in PyTorch [10]. We use the Adam [9] optimizer with a learning rate of 0.05 for latent mapper and 0.002 for StyleGANs. The training process includes 300 iterations for prompt learning and 300 iterations for generator adaptation, using a single NVIDIA RTX 3090 GPU. The batch size is set to 32 for prompt learning and 2 for generator adaptation. The number of learned prompt vectors m is set to 4. The 4 prompt vectors are initialized as the word embeddings of "a photo of a". We use the same Layer-Freezing technique as NADA [2] to select the suitable training layers for each iteration and set the exponential moving average (EMA) decay [17] to 0.99. In the domain regularization loss, following CLIP [13], we separately concatenate 79 manually designed sets of prompts (e.g., "a photo of a ...", "a drawing of a ...") with a domain label and feed them into E_{T}. The average vector of the 79 encoded feature vectors replaces the encoded feature vector of the domain label $E_{\mathrm{T}}(\mathrm{Y_s})$ or $E_{\mathrm{T}}(\mathrm{Y_t})$. For each domain, the ratio parameter λ of the domain regularization loss is selected among [1, 10], according to the best Inception Score [14] of adapted generators. The values of λ on different settings are provided in Tab.1. Compared with NADA, the additional training time from the latent mapper is about 10 minutes, which is easily acceptable.

Diff-IPL. Applied with the Adam [9] optimizer, the learning rates for latent mapper and diffusion autoencoders [12] are set to $7\mathrm{e}^{-2}$ and $3\mathrm{e}^{-5}$, respectively. The training process requires higher memory cost, utilizing a single NVIDIA A6000 GPU. Following Diff-CLIP [8], the batch size is set to 1 for generator adaptation. We also precompute the latent codes of 50 training images via the reverse process of diffusion autoencoders and train target-domain generators for 5 epochs as [8]. We can further accelerate training with fewer diffusion discretization steps [16]. In our experiments, the number of forward steps and reverse steps are reduced to 100 and 250, respectively.

Metrics. We utilize Inception Score (IS) [14], Single Image Fréchet Inception Distance (SIFID) [15], Structural Consistency Score (SCS) [18] and identity similarity (ID) [1, 3] for quantitative evaluation. In specific, for ID, we compute the identity similarity in ArcFace [1] for FFHQ (human faces). For AFHQ (dog faces), we apply TransFG [3], a fine-grained species recognition approach to extract identity features and compute the cosine similarity between source and target (generated) images. For SIFID, we manually collect several reference images of each target domain from the internet and compute the SIFID score for each reference image. We enclose these reference images in the folder "reference". Although the variance of different reference images may lead to an imprecise score in some extreme cases, the superiority of an effective method could still be verified if it outperforms others in most cases.

Table 1. Loss term ratio λ on different settings.

Setting	Source→Target	λ
GAN-IPL	Photo→Disney	1
	Photo→Anime painting	1
	Photo→Wall painting	1
	Photo→Ukiyo-e	1
	Human→Pixar character	1
	Human→Tolkien elf	5
	Human→Werewolf	5
	Photo→Cartoon	10
	Photo→Pointillism	10
	Photo→Cubism	10
Diff-IPL	Photo→Wall painting	3
	Human→Tolkien elf	2

B. Latent Space Interpolation

The state-of-the-art generative models [4–7] all have smooth latent spaces for source-domain image generation. We show that the target-domain generators obtained by our method also preserve this superiority. In Fig.1, each row contains a sequence of images from the same target domain, the left-most column and right-most column are respectively two images $G_{\mathrm{t}}(w_1)$ and $G_{\mathrm{t}}(w_2)$ synthesized with two different latent codes w_1 and w_2. For latent space interpolation, an interpolated image is $G_{\mathrm{t}}((1-\alpha)w_1+\alpha w_2)$, where $\alpha \in [0,1]$. For each row, images from left to right correspond to α ranging from 0 to 1. The visual results show that our method has good robustness and generalization ability.

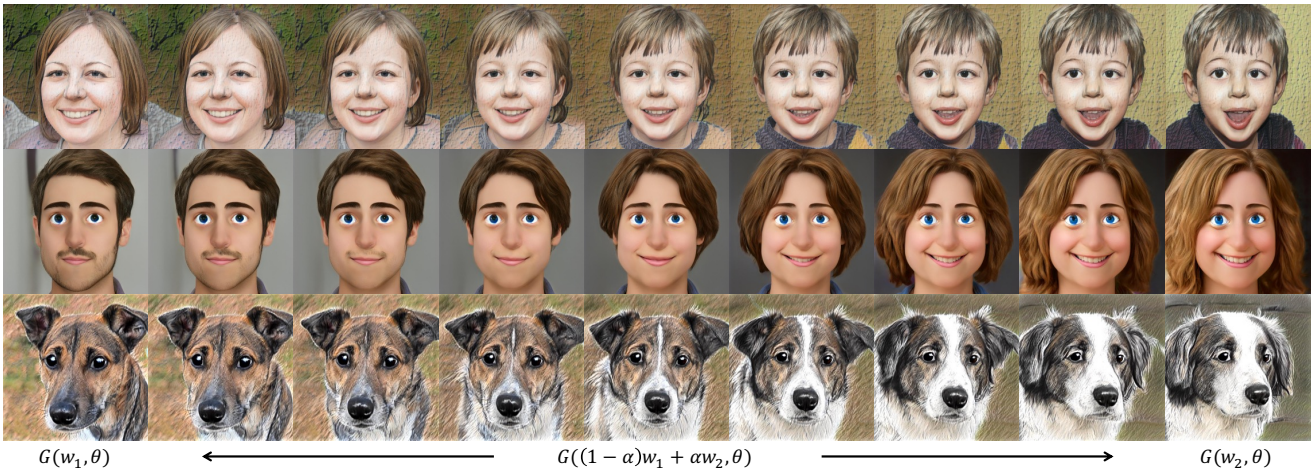

Figure 1. Visual results of latent space interpolation. The source domain is "Photo" while the target domains are "Wall painting", "Pixar character" and "Cartoon" from top to bottom. For each row, the left-most column and right-most column are respectively two images synthesized with two different latent codes. The remaining columns refer to images synthesized with interpolated latent codes.

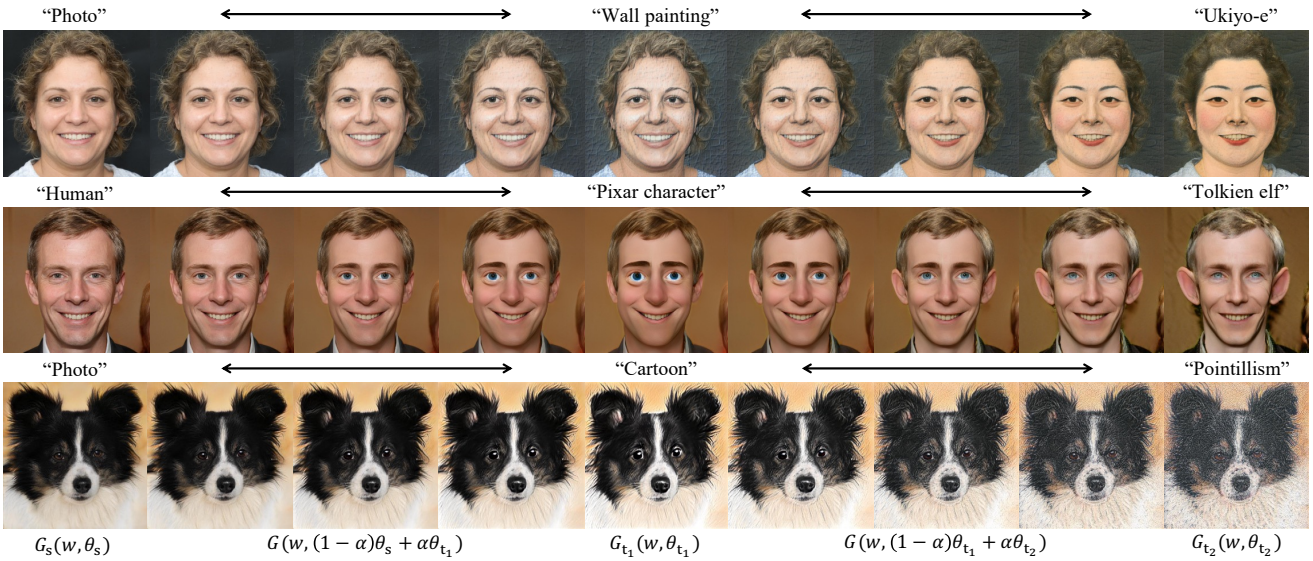

Figure 2. Visual results of cross-model interpolation. In each row, the left-most image is generated by the source-domain generator. The middle and the right-most images are synthesized by two different target-domain generators. The other images represent cross-model interpolations between two different domains.

The various target-domain spaces obtained by our method are consistently smooth.

C. Cross-model Interpolation

Beyond latent space interpolation, we also showcase the model weight smoothness across different domains. In specific, we adopt linear interpolation in weight space for either $G_s(\cdot, \theta_s)$ and $G_t(\cdot, \theta_t)$ or $G_{t_1}(\cdot, \theta_{t_1})$ and $G_{t_2}(\cdot, \theta_{t_2})$, where $G_s(\cdot, \theta_s)$ denotes the source domain generator, $G_{t_1}(\cdot, \theta_{t_1})$ and $G_{t_2}(\cdot, \theta_{t_2})$ denote two adapted generators of different target domains. For example, let θ_1, and θ_2 represent the model weights of two generators. Given a latent code w, we generate the corresponding image by an interpolated model, $G(w, (1-\alpha)\theta_1 + \theta_2)$, where $\alpha \in [0, 1]$. Fig.2 shows that our method has good cross-model interpolation ability, either from a source domain to a target domain or between different target domains.

D. More Well-directed Prompts

A straightforward way to alleviate the mode collapse issue is to manually design a set of well-directed prompts. For example, introduce "with eyes looking forward" as additional prompts to reduce the squinting eyes issue in "Anime painting", or use "with black eyebrows" to solve the blue

eyebrows issue in "Ukiyo-e". In Fig.3, we show that these detailed prompts may lead to other undesired patterns. For "Anime painting", although the squinting eyes issue can be partly addressed, the generated images of NADA shows some similar bleeding eyes patterns. For "Ukiyo-e", the thick black eyebrows replace the original blue eyebrows for generated results of NADA, but the connecting two eyebrows together is a new undesired pattern. It is worth noting that our IPL is still better than NADA with these additional text prompts and avoids undesired patterns.

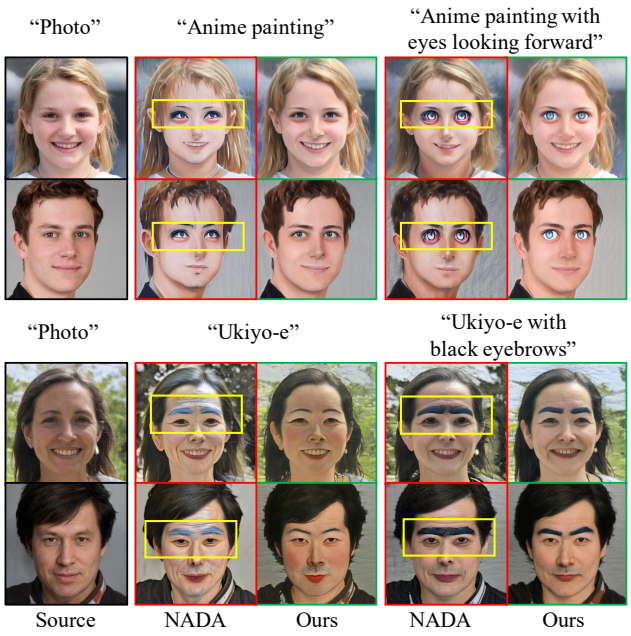

Figure 3. Image synthesis comparison results with more detailed prompts. The source domain is "Photo" and the target domains are "Anime painting" and "Ukiyo-e". Additional prompts are "with eyes looking forward" and "with black eyebrows" for "Anime painting" and "Ukiyo-e", respectively. The yellow box areas show the mode collapse patterns of NADA [2].

E. Geometry Adaptation

As shown in Fig.4, IPL can make diversified geometric edits, such as emotion, haircut, age, and identity like other image manipulation methods.

Figure 4. Geometry adaptation results of IPL.

F. Quantitative Results of Diffusion Models

To quantify the performance improvement of Diff-IPL compared to Diff-CLIP [8] and Diff-CLIP+, IS, SCS, ID and SIFID are evaluated. As illustrated in Tab.2, Diff-IPL performs the best IS, SCS and ID on the two settings, indicating its superiority in the diversity and quality of generated images, together with the structure and identity preservation capability compared to source images. In addition, Diff-IPL achieves the best SIFID score in most cases, showcasing that our method generates the desired target-domain style better.

Table 2. Quantitative evaluation results of Diff-CLIP [8], Diff-CLIP+ and Diff-IPL. S→T, P→WP and H→TE denote Source→Target, Photo→Wall painting and Human→Tolkien elf, respectively. The best results are **bold**.

S→T	Method	IS [14] (↑)	SCS [18] (↑)	ID [1] (↑)	SIFID [15] (↓)		
					R_1	R_2	R_3
P→WP	Diff-CLIP	1.696	0.662	0.595	5.493	5.066	5.727
	Diff-CLIP+	2.542	0.611	0.554	2.644	2.099	2.455
	Diff-IPL	**2.953**	**0.744**	**0.785**	**2.022**	**1.841**	**2.004**
H→TE	Diff-CLIP	2.055	0.684	0.328	6.091	8.138	7.779
	Diff-CLIP+	2.711	0.627	0.399	**2.218**	4.283	4.055
	Diff-IPL	**2.893**	**0.696**	**0.709**	2.749	**3.421**	**3.696**

G. Diffusion Models versus GANs

We compare the Inception Score results of diffusion models and GANs in Fig.3. The superiority of Diff-CLIP+ over NADA indicates diffusion models can handle more cases with better base performance. Assisted with IPL, GAN-IPL showcases competitive performance to Diff-CLIP+. Moreover, integrating IPL with Diff-CLIP+ as Diff-IPL also leads to a significant improvement, indicating IPL's compatibility with both GANs and diffusion models.

Table 3. Quantitative comparison of GANs and diffusion models. We compare the Inception Score (↑) [14] results for Photo→Wall painting and Human→Tolkien elf.

Source→Target	NADA (GAN)	GAN-IPL	Diff-CLIP+	Diff-IPL
Photo→Wall painting	2.183	2.676	2.542	2.953
Human→Tolkien elf	2.479	2.778	2.711	2.893

H. Effect of the Number of Prompts

To ensure comparison fairness, we adopted the setting $m = 4$ in experiments. In Tab.4, we investigate the effect of m by setting it as 1,2,4,8,16, with the same 300 training iterations. The Inception Score results show that learned prompts (results of different m) consistently exceed the manual prompts (NADA). In addition, too small or too

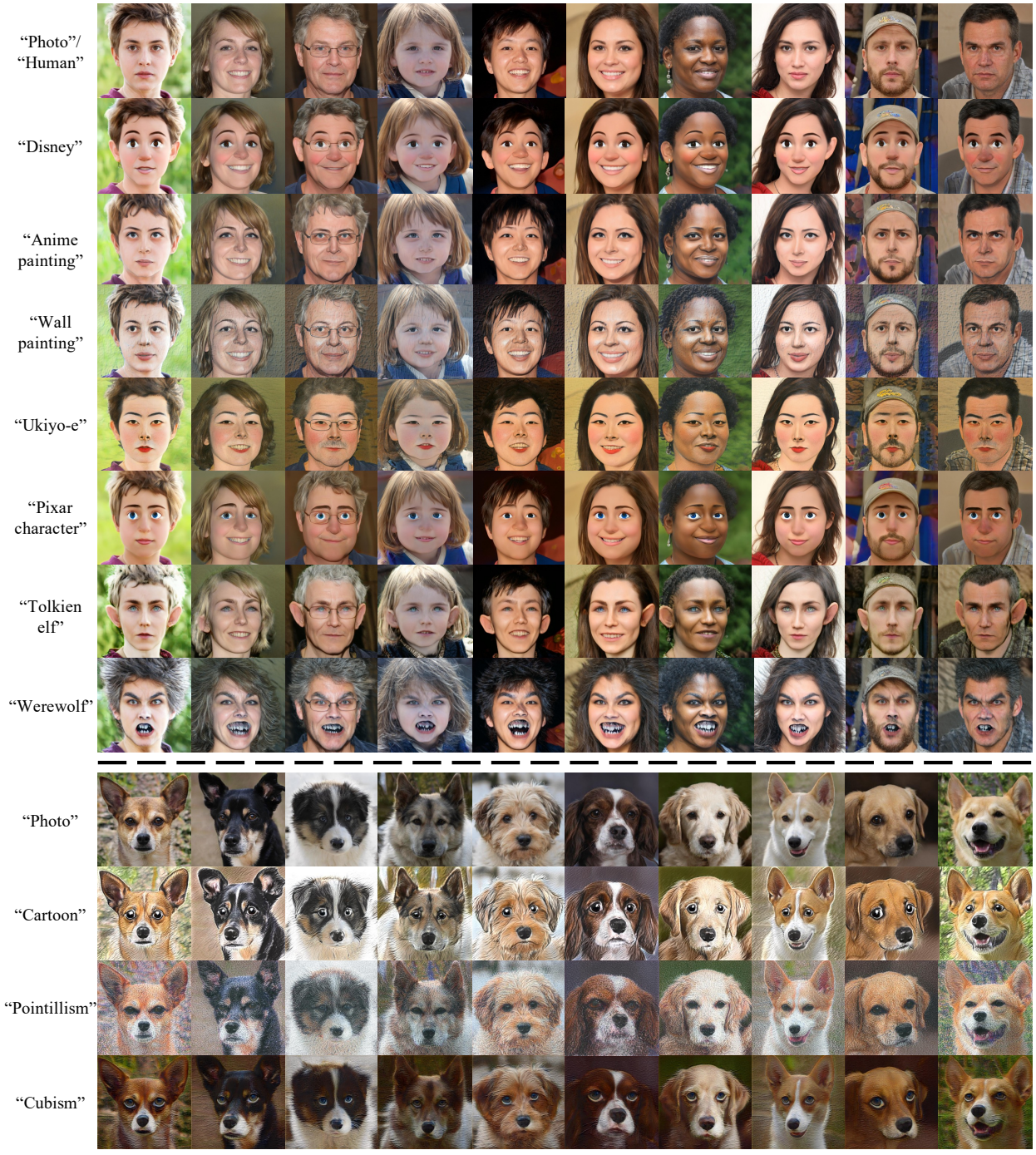

Figure 5. Additional results of GAN-IPL.

large m may lead to insufficient learning and performance degradation. Overall, $m \in [4, 8]$ can be optimal.

Table 4. Quantitative results of different m. We evaluate the Inception Score (↑) [14] for Photo→Ukiyo-e.

Source→Target	NADA	$m = 1$	$m = 2$	$m = 4$	$m = 8$	$m = 16$
Photo→Ukiyo-e	2.205	2.757	2.943	2.974	3.047	2.651

I. More Visual Results

We provides more visual results of GAN-IPL and Diff-IPL across all target domains mentioned in our main paper. In specific, we display additional generative model adaptation results for GAN-IPL in Fig.5 and real-world image translation results for Diff-IPL in Fig.6. Although Diff-IPL has stronger inversion capability for real images (discussed

Figure 6. Additional results of Diff-IPL.

in our main paper), the visual results of GAN-IPL and Diff-IPL seem to be comparable for general cases. In practice, GAN-IPL is more suitable for applications where plenty of target-domain images are required, since GANs perform a more efficient generative process than diffusion models. While Diff-IPL is more appropriate for applications where the structure and identity of source-domain images need to be precisely preserved in target-domain images.

J. Prompt Visualization

Since the prompt vectors are continually optimized, there is no one-to-one correspondence between learned prompt vectors and realistic words. Even so, we try to find some relationships by searching the closest word within the vocabulary for every prompt vector. Following CoOp [19],the Euclidean distance between a prompt vector and the embedding of the a realistic word is computed. We present several cases of these searched image-specific words in Fig.7. Overall, our discovery is similar to the discussion in [19]. A few words are somewhat relevant to their corresponding image, e.g., fashionista, thinkers and musician, while most of the words remain difficult for us to find their connection to images. We conjecture that a source image should contain rich and diverse image-specific semantics. With the limited prompt length, one prompt vector may contain an integration of many different semantics and can not be correctly interpreted with the closest word in the existing vocabulary.

Limitation. Sincerely, the unknown visualization of learned prompt vectors may somewhat limit the interpretability of IPL. We expect that future works could investigate better solutions to effectively decouple and visualize the semantics of a continually optimized prompt vector.

fashionista	musician	alvarez	thrive
dette	redcarpet	capping	speeches
thinkers	howe	graphic	ireland
coutinho	mammoth	boulder	nunes

Figure 7. Visualization of the learned prompt vectors. For each image, we present the nearest words of the prompt vectors computed in the word embedding space. Red words may be somewhat relevant to corresponding images.

K. Large Domain Shift.

In general, there is a strong correlation between source and target domains in domain adaptation tasks. As demonstrated in Fig.8, generator adaptation with a large domain shift (e.g., from "Human" to "Cat") is challenging for all existing zero-shot generators and requires future investigation. However, we can observe that IPL could present more

Figure 8. Image synthesis comparison results with a large domain shift. The source domain is "Human" and the target domain is "Cat". We compare IPL with StyleCLIP [11] and NADA [2].

cat-like whiskers and eyes, compared with other zero-shot competitors, i.e., StyleCLIP [11] and NADA [2].

L. Social Impact

IPL may contribute to artistic image synthesis applications in social media industries. It may also assist the other computer vision tasks (e.g., recognition and detection) as a data augmentation technique. However, the ability of IPL to synthesize fake images from real-world images may bring some ethical problems, which must be treated carefully.

References

[1] Jiankang Deng, Jia Guo, Niannan Xue, and Stefanos Zafeiriou. ArcFace: Additive angular margin loss for deep face recognition. In *CVPR*, 2019. 1, 3

[2] Rinon Gal, Or Patashnik, Haggai Maron, Amit H Bermano, Gal Chechik, and Daniel Cohen-Or. StyleGAN-NADA: CLIP-guided domain adaptation of image generators. In *SIGGRAPH*, 2022. 1, 3, 6

[3] Ju He, Jie-Neng Chen, Shuai Liu, Adam Kortylewski, Cheng Yang, Yutong Bai, and Changhu Wang. TransFG: A transformer architecture for fine-grained recognition. In *AAAI*, 2022. 1

[4] Tero Karras, Miika Aittala, Janne Hellsten, Samuli Laine, Jaakko Lehtinen, and Timo Aila. Training generative adversarial networks with limited data. In *NeurIPS*, 2020. 1

[5] Tero Karras, Miika Aittala, Samuli Laine, Erik Härkönen, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Alias-free generative adversarial networks. In *NeurIPS*, 2021. 1

[6] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *CVPR*, 2019. 1

[7] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of StyleGAN. In *CVPR*, 2020. 1

[8] Gwanghyun Kim, Taesung Kwon, and Jong Chul Ye. DiffusionCLIP: Text-guided diffusion models for robust image manipulation. In *CVPR*, 2022. 1, 3

[9] Diederik P Kingma and Jimmy Ba. ADAM: A method for stochastic optimization. In *ICLR*, 2015. 1

[10] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. PyTorch: An imperative style, high-performance deep learning library. In *NeurIPS*, 2019. 1

[11] Or Patashnik, Zongze Wu, Eli Shechtman, Daniel Cohen-Or, and Dani Lischinski. StyleCLIP: Text-driven manipulation of StyleGAN imagery. In *ICCV*, 2021. 6

[12] Konpat Preechakul, Nattanat Chatthee, Suttisak Wizadwongsa, and Supasorn Suwajanakorn. Diffusion Autoencoders: Toward a meaningful and decodable representation. In *CVPR*, 2022. 1

[13] 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. 1

[14] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training GANs. In *NeurIPS*, 2016. 1, 3, 4

[15] Tamar Rott Shaham, Tali Dekel, and Tomer Michaeli. SinGAN: Learning a generative model from a single natural image. In *ICCV*, 2019. 1, 3

[16] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *ICLR*, 2020. 1

[17] Antti Tarvainen and Harri Valpola. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. In *NeurIPS*, 2017. 1

[18] Jiayu Xiao, Liang Li, Chaofei Wang, Zheng-Jun Zha, and Qingming Huang. Few shot generative model adaption via relaxed spatial structural alignment. In *CVPR*, 2022. 1, 3

[19] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Learning to prompt for vision-language models. *IJCV*, 2022. 5