Formal Definition of Fingerprints Improves Attribution of Generative Models

Hae Jin Song[1,2] * **Mahyar Khayatkhoei**[1] **Wael AbdAlmageed**[1,2,3]
[1]USC Information Sciences Institute, Marina del Rey, USA
[2]USC Thomas Lord Department of Computer Science, Los Angeles, USA
[3]USC Ming Hsieh Department of Electrical and Computer Engineering, Los Angeles, USA

Abstract

Recent works have shown that generative models leave traces of their underlying generative process on the generated samples, broadly referred to as fingerprints of a generative model, and have studied their utility in detecting synthetic images from real ones. However, the extent to which these fingerprints can distinguish between various types of synthetic images and identify the underlying generative process remain under-explored. In particular, the very definition of a fingerprint remains unclear, to our knowledge. To that end, in this work, we formalize the definition of artifact and fingerprint in generative models, propose an algorithm for computing them in practice, and study how different design parameters affect the fingerprints and their attributability. We find that using our proposed definitions can significantly improve the performance on the task of identifying the underlying generative process from samples (model attribution) compared to existing methods. Additionally, we study the structure of the fingerprints and observe that it is very predictive of the effect of different design choices on the generative process.

1 Introduction

Recent years have seen rapid developments of generative models and their integration into our society. However, there is still a big gap in understanding what makes these models behave the way they do and, in particular, how different choices in designing generative models (*e.g*. model architecture, training data, training objectives and optimization parameters) contribute to their different behaviors. In this work, we address this question by considering generative models' behaviors through the lens of artifacts and fingerprints they leave on their samples. In other words, we investigate how different design parameters affect model fingerprints and their attributability back to the design choices. Our work focuses on three key design parameters: model architecture, learning algorithm, and training datasets. We evaluate their effects on model attribution by measuring the attributability of a generated image to the underlying design factors. We consider the following three cases:

- Effects of model-type: how the choice of models – as a combination of model architecture and learning algorithm, *e.g*. StyleGAN3 vs. NVAE – affects fingerprints and attribution
- Effects of training data on fingerprints, independently of the choice of model-type
- Effects of layer types (*e.g*. type of upsampling, non-linearity, normalization) on fingerprints

We systematically explore these questions by (1) proposing formal definitions of artifacts and fingerprints of generative models, (2) formulating a manifold-based attribution process using our definitions to predict the design factors of generated samples (*i.e*. images), and (3) conducting extensive experiments on a large array of generative models, including state-of-the-art models.

In summary, our contributions are as follows:

*Corresponding author: Hae Jin Song <haejinso@usc.edu>

ATTRIB: Workshop on Attributing Model Behavior at Scale at NeurIPS 2023.

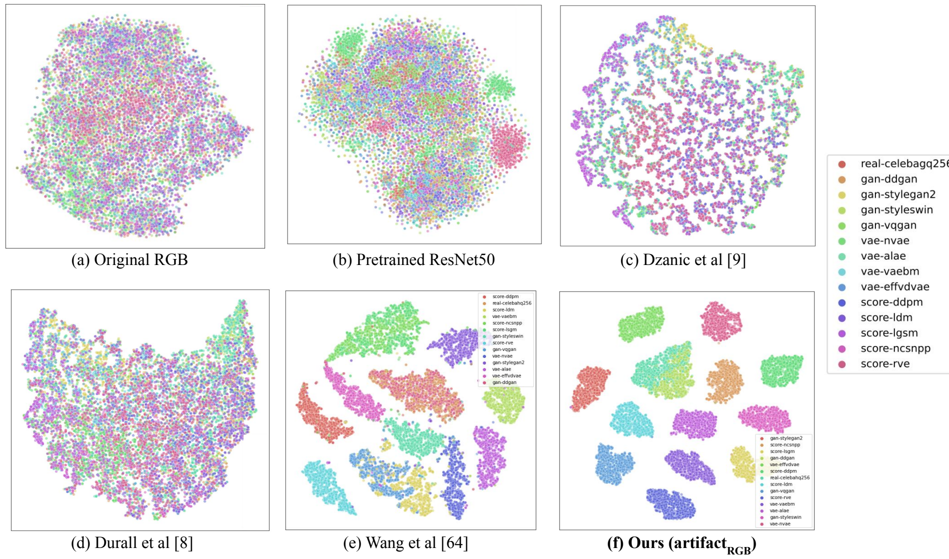


Figure 1: Features learned using our definition of artifacts(f) achieve better separation between samples from different generative models (shown in *different colors*). (a) shows tSNE of generated samples in pixel space, (b) in the feature space of ResNet50 pretrained on ImageNet, and (c-f) in the penultimate layer of the classifier proposed by each baseline method trained on the task of model-type attribution.

- We propose formal definitions of artifacts and fingerprints of generative models that have been missing in the literature, and provide an algorithm to compute them from finite samples.
- We theoretically justify our definition by relating it to two prominent metrics for generative models: Precision and Recall (39; 25) and Integral Probability Metrics (IPMs) (33; 41).
- We use our definitions to evaluate the effects of three main factors in designing generative models (model architecture, learning algorithms, training dataset) on model fingerprints.
- We conduct extensive experiments to show the effectiveness of our fingerprints in distinguishing generative models, outperforming existing attribution methods. In particular, our experiments consider a large array of generative models from all four main families (GAN, VAE, Flow, Score-based) as opposed to a small number of GANs or VAEs in exiting works.
- Our results show that each design factor (dataset, learning algorithm and model architecture) independently contributes to identifiable artifacts in generated images and, in particular, the type of loss function and upsampling has the most significant effect on the fingerprints. These findings confirm the general intuition in the research community about the sources of limitations in generative models (6; 7), thereby supporting the utility of our definitions.

2 ManiFPT: manifold-based fingerprints of generative models

We approach the study of generative models' behaviors and their attribution to underlying design factors by looking at the artifacts the models leave on their samples. However, as discussed in Sec. 1 and Sec. B, despite the existing works that suggest evidence for their existence, concrete definitions of these terms remain unclear. Therefore, in this section, we first motivate and propose formal definitions for the artifacts and fingerprints of generative models, and describe an algorithm for computing them from observed samples. We then use our definitions to formulate a new manifold-based attribution that predicts the design factors of generative models from their samples.

2.1 Definitions of GM artifacts and fingerprints

Intuitively, the artifacts and fingerprints of generative models are the traces of their imperfection in modeling the true data-generating process, which can be extracted from the samples they generate. In the framework of manifold learning (2; 9), which hypothesizes that many high-dimensional real-world datasets (*e.g.* images and videos) lie on a lower-dimensional manifold, we formalize such *imperfections* of generative models as the deviation of generated samples from the true data manifold (*i.e. artifact*). More concretely, consider a generative model G trained on a dataset X of samples that lie on a true data manifold $\mathcal{M}$. Let P denote the true data distribution and Q the induced probability distribution of G with the support of S_G. Let x_G be a sample generated by G:

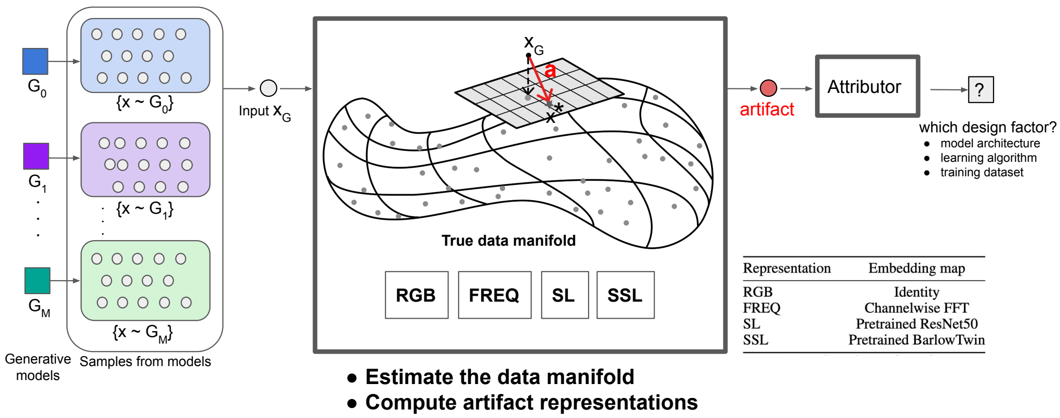


Figure 2: **Our attribution method.** We propose an attribution method based on our definition of artifacts: Given input images X_G, we first map them to an embedding space (RGB, Frequency, feature space of a pretrained supervised-learning (SL) network, or of a pretrained self-supervised learning (SSL) network), and compute their artifacts a as defined in Sec. 2.1. We train the attributor (classifier) to predict the design factor of the source generative model, under the standard cross-entropy loss.

Definition 2.1 (Artifact)**.** An **artifact** of a model-generated sample x_G with respect to the data manifold $\mathcal{M}$ is defined as the difference between x_G and its closest point on the data manifold (x^*). That is, given a data manifold $\mathcal{M}$ equipped with a distance metric $d_{\mathcal{M}}$:

$$x^* := \operatorname*{argmin}_{x \in \mathcal{M}} d_{\mathcal{M}}(x_G, x) \tag{1}$$

$$a_{\mathcal{M}}(x_G) := x_G - x^* \tag{2}$$

Definition 2.2 (Fingerprint)**.** The **fingerprint** (**FPT**) of a generative model G, with respect to the true data manifold $\mathcal{M}$, is defined as the set of all its feasible artifacts:

$$\text{FPT}_G = \{a_{\mathcal{M}}(x) | x \in S_G\} \tag{3}$$

2.2 Estimation of GM artifacts and fingerprints

Estimating the data manifold. Since in practice we have only finite samples, rather than the entire manifold, we estimate the above definitions by taking a minimum over the set of observed samples: we use real images in the training datasets of the generative models, and map them to a suitable embedding space to construct a collection of features to be used as an estimated image manifold. One key modeling decision in this step is the choice of an embedding space for the image manifold. Ideally, the embedding space should capture meaningful fingerprint features of generative models. We consider four spaces based on the previous works that suggest the existences of fingerprints (31; 7) and visual features (42; 51) encoded in their representations: RGB, Frequency, and feature spaces learned by a supervised-learning method (SL) (*e.g.* ResNet50 (13)) and by a self-supervised learning method (SSL) (*e.g.* Barlow Twins (51)). See. Appendix. E.1 for more details.

Computing the artifacts An artifact of a model-generated sample x_G is computed in two steps:

1. Estimate the projection $x^\star$ by minimizing the distance to x_G over the points in X_M. We use the Euclidean distance, *i.e.* $d_M(x, x_G) := ||x - x_G||^2$
2. Compute the artifact as difference, $a(x_G) = x_G - x^*$

See Fig. 3 for examples of the artifacts computed in this way.

Fingerprint of a model Given a finite set of model generated samples $X_G = \{x_i\}_{i=1}^N$ where $x_i \sim Q$, we estimate its fingerprint by computing an artifact of each sample in X_G.

2.3 Attribution of GM artifacts to design parameters

Based on our definitions, we now propose a new method for predicting the design parameters of generative models from their samples (Fig.2). Our system consists of two modules, the artifact-representation module and the attribution module. The first module represents an input image as an

Table 1: **Model-type attribution results.** We evaluate artifact features on the task of predicting the model-types of generated images. Separability of the feature spaces are measured in FD ratio (FDR). Higher FDR means better separability. Our methods based on the proposed definition of artifacts outperform all baseline methods.

	GM-CelebA		GM-CHQ		GM-FFHQ	
Methods	Acc.(%)↑	FDR↑	Acc.(%)↑	FDR↑	Acc.(%)↑	FDR↑
McClo18(31)	62.6 ± 2.314	70.2	57.4 ± 2.013	36.3	50.8 ± 0.341	26.3
Nataraj19 (34)	61.1 ± 2.203	74.0	56.3 ± 1.325	37.9	51.3 ± 0.581	35.3
Durall20 (6)	62.2 ± 2.243	75.5	59.1 ± 1.301	38.8	60.9 ± 0.255	37.9
Dzanic20 (7)	61.6 ± 2.029	88.1	56.9 ± 1.215	38.2	55.7 ± 0.324	30.3
Wang20 (47)	62.2 ± 1.203	89.8	59.5 ± 1.252	30.3	64.2 ± 0.310	37.9
Marra18 (29)	63.1 ± 1.103	83.4	51.3 ± 1.281	20.5	53.2 ± 0.218	30.4
Marra19 (30)	61.1 ± 1.729	101.4	59.1 ± 1.27	34.9	51.8 ± 0.233	30.9
Yu2019 (50)	60.6 ± 1.103	111.4	61.1 ± 1.122	<u>74.5</u>	60.5 ± 0.105	35.1
$Ours_{\text{RGB}}$	70.5 ± 1.565	115.3	<u>63.7</u> ± 1.238	64.2	<u>65.3</u> ± 0.125	<u>50.1</u>
$Ours_{\text{FREQ}}$	72.8 ± 1.321	120.9	**64.8** ± 1.124	70.1	**66.1** ± 0.207	**57.6**
$Ours_{\text{SL}}$	<u>73.6</u> ± 1.102	**168.0**	62.3 ± 1.221	**77.2**	63.2 ± 0.305	49.8
$Ours_{\text{SSL}}$	**74.7** ± 1.121	<u>125.9</u>	61.9 ± 1.351	63.3	63.8 ± 0.203	40.9

artifact feature by following the algorithm in Sec 2.2.The second module passes the artifact features to a classifier and predicts the design factors of the source generative models.

Training our attribution network We use the pretrained ResNet50 (13) as the backbone of our classifier, attach a new softmax layer and fine-tune it under the standard cross-entropy loss.

2.4 Theoretical properties of GM artifacts and fingerprints

Our proposed definitions of GM artifacts and fingerprints are closely related to two prominent metrics on generative models: Precision and Recall (P&R) (39; 25) and integral probability metrics (IPMs)(33; 41). The most fundamental relation is that under our definition, the fingerprint is non-zero if and only if two distributions have unequal supports. See Appendix A for full proofs and discussions.

3 Experiments and Results

We evaluate the attributability of generated images and their artifacts to the design parameters by varying the following parameters independently: model-type (Sec. 3.1,F.1), training data (Sec. F.2), and layer type (Sec. F.3). See Appendix. E for details on our experimental setup (E.2) and results(F).

3.1 Effect of model-type on fingerprints and their attributability

We investigate how the model-type (*i.e.* a combination of model architecture and learning algorithm) affects model fingerprints and their attributability. To do so, we measure the accuracy of attributing artifacts of generative models, trained on the *same* training data, to the model-type of the source models (*e.g.* StyleGAN2 vs. VQ-GAN vs Latent Diffusion Model vs etc.). To complement the accuracy, we measure the separability of fingerprint representations using the ratio of inter-class and intra-class Fréchet Distance (FDR) (5). The larger the ratio, the more attributable the fingerprints are to their model-type. See Appendix F.4 for the definition of FDR and how to compute it. **Results.** Tab. 1 shows the results of model-type attribution in accuracy and FDR. First, the choice of model-type (BigGAN vs. StyleGAN2 vs. NVAE etc.) makes attributable artifacts in both color (McClo18, Nataraj19) and frequency (*e.g.* Durall20, Dzanic20), as well as in our artifact representations. This result is consistent with the exiting observations in the literature and suggests our proposed definitions capture the notion of artifacts as desired. Furthermore, our proposed methods outperforms all baselines by meaningful margins: 11.6%, 3.7%, 1.9% in each dataset, and 5.73% on average. This indicates that our methods better capture features that differentiate one generative model from another, further supporting our definitions' usefulness as fingerprints of generative models.

See Appendix. F for experiments on the effects of training data and layer types on model attribution.

4 Conclusion

Our work addresses the problem of understanding generative models and the design factors that contribute to their different behaviors. To do so in a principled way, we formally define artifacts and fingerprints and study the effects of model-types, training data and layer types on the fingerprints via attribution tasks. Importantly, we show that our proposed definition outperforms existing methods on model attribution and provides a useful feature space for differentiating various generative models.

Acknowledgments and Disclosure of Funding

This research was funded by the US Government (DARPA SemaFor project HR0011046421). The views, and conclusions contained herein are those of the author(s) and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the Air Force Laboratory (AFRL), DARPA, or the U.S. Government.

References

[1] Andrew Brock, Jeff Donahue, and Karen Simonyan. Large Scale GAN Training for High Fidelity Natural Image Synthesis. In *International Conference on Learning Representations*, Jan. 2023.

[2] Lawrence Cayton. Algorithms for manifold learning. *Univ. of California at San Diego Tech. Rep*, 12(1-17):1, 2005.

[3] Keshigeyan Chandrasegaran, Ngoc-Trung Tran, and Ngai-Man Cheung. A Closer Look at Fourier Spectrum Discrepancies for CNN-generated Images Detection, Mar. 2021. arXiv:2103.17195 [cs, eess].

[4] Ricky T. Q. Chen, Jens Behrmann, David K Duvenaud, and Joern-Henrik Jacobsen. Residual Flows for Invertible Generative Modeling. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

[5] DC Dowson and BV666017 Landau. The fréchet distance between multivariate normal distributions. *Journal of multivariate analysis*, 12(3):450–455, 1982.

[6] R. Durall, Margret Keuper, and J. Keuper. Watch Your Up-Convolution: CNN Based Generative Deep Neural Networks Are Failing to Reproduce Spectral Distributions. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[7] Tarik Dzanic, Karan Shah, and Freddie Witherden. Fourier Spectrum Discrepancies in Deep Network Generated Images. In *Advances in Neural Information Processing Systems*, volume 33, pages 3022–3032. Curran Associates, Inc., 2020.

[8] Patrick Esser, Robin Rombach, and Bjorn Ommer. Taming Transformers for High-Resolution Image Synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12873–12883, 2021.

[9] Charles Fefferman, Sanjoy Mitter, and Hariharan Narayanan. Testing the manifold hypothesis. *Journal of the American Mathematical Society*, 29(4):983–1049, 2016.

[10] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron C. Courville, and Yoshua Bengio. Generative adversarial nets. In *NIPS*, 2014.

[11] Ishaan Gulrajani, Faruk Ahmed, Martín Arjovsky, Vincent Dumoulin, and Aaron C. Courville. Improved training of wasserstein gans. In *NIPS*, 2017.

[12] Louay Hazami, Rayhane Mama, and Ragavan Thurairatnam. Efficient-VDVAE: Less is more, Apr. 2022.

[13] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[14] Irina Higgins, Loïc Matthey, Arka Pal, Christopher P. Burgess, Xavier Glorot, Matthew M. Botvinick, Shakir Mohamed, and Alexander Lerchner. beta-vae: Learning basic visual concepts with a constrained variational framework. In *ICLR*, 2017.

[15] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising Diffusion Probabilistic Models. In *Advances in Neural Information Processing Systems*, volume 33, pages 6840–6851. Curran Associates, Inc., 2020.

[16] Xianxu Hou, L. Shen, Ke Sun, and Guoping Qiu. Deep feature consistent variational autoencoder. *2017 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1133–1141, 2017.

[17] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of gans for improved quality, stability, and variation. *ArXiv*, abs/1710.10196, 2018.

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

[19] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4396–4405, 2019.

[20] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of stylegan. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8107–8116, 2020.

[21] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and Improving the Image Quality of StyleGAN. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8107–8116, Seattle, WA, USA, June 2020. IEEE.

[22] 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 *Proceedings of the 39th International Conference on Machine Learning*, pages 11201–11228. PMLR, June 2022.

[23] Durk P Kingma and Prafulla Dhariwal. Glow: Generative Flow with Invertible 1x1 Convolutions. In *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018.

[24] Naveen Kodali, Jacob Abernethy, James Hays, and Zsolt Kira. On convergence and stability of gans. *arXiv preprint arXiv:1705.07215*, 2017.

[25] Tuomas Kynkäänniemi, Tero Karras, Samuli Laine, Jaakko Lehtinen, and Timo Aila. Improved precision and recall metric for assessing generative models. *Advances in Neural Information Processing Systems*, 32, 2019.

[26] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proceedings of International Conference on Computer Vision (ICCV)*, December 2015.

[27] Xuezhe Ma and Eduard H. Hovy. Macow: Masked convolutional generative flow. In *NeurIPS*, 2019.

[28] Xudong Mao, Qing Li, Haoran Xie, Raymond Y. K. Lau, Zhen Wang, and Stephen Paul Smolley. Least squares generative adversarial networks. *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 2813–2821, 2017.

[29] Francesco Marra, Diego Gragnaniello, Davide Cozzolino, and Luisa Verdoliva. Detection of GAN-Generated Fake Images over Social Networks. In *2018 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR)*, pages 384–389, Apr. 2018.

[30] Francesco Marra, Cristiano Saltori, Giulia Boato, and Luisa Verdoliva. Incremental learning for the detection and classification of GAN-generated images. In *2019 IEEE International Workshop on Information Forensics and Security (WIFS)*, pages 1–6, Dec. 2019.

[31] Scott McCloskey and Michael Albright. Detecting GAN-generated Imagery using Color Cues, Dec. 2018.

[32] Aaron F McDaid, Derek Greene, and Neil Hurley. Normalized mutual information to evaluate overlapping community finding algorithms. *arXiv preprint arXiv:1110.2515*, 2011.

[33] Alfred Müller. Integral Probability Metrics and Their Generating Classes of Functions. *Advances in Applied Probability*, 29(2):429–443, 1997.

[34] Lakshmanan Nataraj, Tajuddin Manhar Mohammed, Shivkumar Chandrasekaran, Arjuna Flenner, Jawadul H. Bappy, Amit K. Roy-Chowdhury, and B. S. Manjunath. Detecting GAN generated Fake Images using Co-occurrence Matrices, Oct. 2019.

[35] Ehsan Nowroozi, Mauro Conti, and Yassine Mekdad. Detecting High-Quality GAN-Generated Face Images using Neural Networks, Mar. 2022.

[36] Stanislav Pidhorskyi, Donald A. Adjeroh, and Gianfranco Doretto. Adversarial Latent Autoencoders. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 14092–14101, Seattle, WA, USA, June 2020. IEEE.

[37] Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. *CoRR*, abs/1511.06434, 2016.

[38] 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.

[39] Mehdi SM Sajjadi, Olivier Bachem, Mario Lucic, Olivier Bousquet, and Sylvain Gelly. Assessing generative models via precision and recall. *Advances in neural information processing systems*, 31, 2018.

[40] Yang Song, Jascha Sohl-Dickstein, Diederik P. Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-Based Generative Modeling through Stochastic Differential Equations. In *International Conference on Learning Representations*, Jan. 2023.

[41] Bharath K. Sriperumbudur, Kenji Fukumizu, Arthur Gretton, Bernhard Schölkopf, and Gert R. G. Lanckriet. On integral probability metrics, \phi-divergences and binary classification, Oct. 2009.

[42] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the Inception Architecture for Computer Vision. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2818–2826, Las Vegas, NV, USA, June 2016. IEEE.

[43] Arash Vahdat and Jan Kautz. Nvae: A deep hierarchical variational autoencoder. *ArXiv*, abs/2007.03898, 2020.

[44] Arash Vahdat and Jan Kautz. NVAE: A Deep Hierarchical Variational Autoencoder. In *Advances in Neural Information Processing Systems*, volume 33, pages 19667–19679. Curran Associates, Inc., 2020.

[45] Arash Vahdat, Karsten Kreis, and Jan Kautz. Score-based Generative Modeling in Latent Space. In *Advances in Neural Information Processing Systems*, volume 34, pages 11287–11302. Curran Associates, Inc., 2021.

[46] Laurens van der Maaten and Geoffrey E. Hinton. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9:2579–2605, 2008.

[47] Sheng-Yu Wang, O. Wang, Richard Zhang, Andrew Owens, and Alexei A. Efros. CNN-Generated Images Are Surprisingly Easy to Spot… for Now. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[48] Zhisheng Xiao, Karsten Kreis, Jan Kautz, and Arash Vahdat. VAEBM: A Symbiosis between Variational Autoencoders and Energy-based Models. In *International Conference on Learning Representations*, Feb. 2022.

[49] Zhisheng Xiao, Karsten Kreis, and Arash Vahdat. TACKLING THE GENERATIVE LEARNING TRILEMMA WITH DENOISING DIFFUSION GANS. 2022.

[50] Ning Yu, Larry Davis, and Mario Fritz. Attributing Fake Images to GANs: Learning and Analyzing GAN Fingerprints, Aug. 2019.

[51] Jure Zbontar, Li Jing, Ishan Misra, Yann LeCun, and Stéphane Deny. Barlow twins: Self-supervised learning via redundancy reduction. *arXiv preprint arXiv:2103.03230*, 2021.

[52] Bowen Zhang, Shuyang Gu, Bo Zhang, Jianmin Bao, Dong Chen, Fang Wen, Yong Wang, and Baining Guo. StyleSwin: Transformer-based GAN for High-resolution Image Generation. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11294–11304, New Orleans, LA,

USA, June 2022. IEEE.

A Proofs on relationships between fingerprints and P&R and IPMs

Our proposed definitions of GM artifacts and fingerprints are closely related to two prominent metrics for distinguishing generative models: Precision and Recall (P&R) (39; 25) and integral probability metrics (IPMs)(33; 41). The most fundamental relation is that under our definition, the fingerprint is non-zero if and only if two distributions have unequal supports. From this fact several properties of fingerprints under our definition readily follow:

(A) Relation to Precision and Recall (P&R) Let P denote the true data distribution, Q the generator G's distribution, and $\text{FPT}(Q;P)$ the fingerprint of G w.r.t. P as defined in 2.1 Let $d_{\text{FPT}}(Q;P)$ be the size of a largest artifact vector in $\text{FPT}(Q;P)$ defined as,

$$d_{\text{FPT}}(Q;P) := \sup_{x_G \sim Q} \{||a||_2 : a \in FPT(Q;P)\} \tag{4}$$

Informally, $d_{\text{FPT}}(Q;P)$ is one way to quantify the maximal deviation of the generator's manifold (i.e., $\text{Supp}(Q)$) from the data manifold (i.e., $\text{Supp}(P)$). Note that $d_{\text{FPT}}(Q;P) \geq 0$. First of all, the following equivalences hold:

$$\begin{aligned} &\text{"All images } x_G \text{ from } G \text{ lie on the true data manifold"} \\ &\Leftrightarrow \forall x_G \sim Q : x_G \in \text{Supp}(P) \Leftrightarrow x^\star = x_G && (5) \\ &\Leftrightarrow \forall x_G \sim Q : a(x_G) = \vec{0} \quad \text{(by Eqn.2)} && (6) \\ &\Leftrightarrow \text{FPT}(Q;P) = \{\vec{0}\} && (7) \\ &\Leftrightarrow d_{\text{FPT}}(Q;P) = 0 && (8) \end{aligned}$$

By the definition of P&R in Defn (2) of (25),

$$\forall x_G \sim Q : x_G \in \text{Supp}(P) \Leftrightarrow \text{Precision}(Q,P) = 1 \tag{9}$$

Therefore, $d_{\text{FPT}}(Q;P) = 0 \Leftrightarrow \text{Precision}(Q,P) = 1$, and the minimum achievable deviation of Q from P based on our definitions of artifacts and fingerprints corresponds to the maximal achievable precision.

Similarly, by considering FPT of P with respect to Q where Q is now the reference distribution,

$$\begin{aligned} \text{FPT}(P;Q) = \{\vec{0}\} &\Leftrightarrow d_{\text{FPT}}(P;Q) = 0 && (10) \\ &\Leftrightarrow \text{Recall}(Q;P) = 1 && (11) \end{aligned}$$

In other words, the minimum achievable deviation of P from Q corresponds to the maximal recall.

In summary, the following relationships between our definition of fingerprint and P&R hold:

$$\text{FPT}(Q;P) = \{\vec{0}\} \Leftrightarrow \text{Precision}(Q,P) = 1 \ \textbf{(max precision)} \tag{12}$$

$$\text{FPT}(P;Q) = \{\vec{0}\} \Leftrightarrow \ \text{Recall}(Q,P) = 1 \ \textbf{(max recall)} \tag{13}$$

Additionally, we have the property of equal supports:

$$\begin{aligned} &\text{FPT}(Q;P) = \{\vec{0}\} \ \text{ and } \ \text{FPT}(P;Q) = \{\vec{0}\} \\ \Leftrightarrow \quad &\text{Supp(P)} = \text{Supp}(Q) \quad \textbf{(equal supports)} && (14) \end{aligned}$$

The property of equal supports implies the degree to which our fingerprint is able to capture the difference between P and Q: as long as there is at least one generated datapoint that does not lie on the data manifold, our fingerprint (either Q w.r.t P or P w.r.t Q) can encode that difference by having at least one non-zero element and its d_{FPT} strictly greater than zero.

(B) Relation to integral probability metrics (IPMs) Our definition of Fingerprint is related to integral probability metrics (IPMs)(33; 41), which include MMD and Wasserstein distance, in the following way: By the property of equal supports above,

$$\text{Supp}(P) \neq \text{Supp}(Q) \Leftrightarrow \exists a \in \text{FPT}(Q;P) \neq \vec{0} \tag{15}$$

By the definition of IPMs (33; 41),

$$\text{Supp}(P) \neq \text{Supp}(Q) \Rightarrow \exists \text{IPM}(Q,P) \neq 0 \tag{16}$$

Table 2: **Our dataset of generation models.** We create three new benchmark datasets for (generative model) attribution task by collecting samples from a large variety of models, trained on CelebA (26), CelebA-HQ (CHQ) (17) and FFHQ (17).

Family	GM-CelebA	GM-CHQ	GM-FFHQ
Real	CelebA (26)	CelebA-HQ (256) (17)	FFHQ (256) (19)
GAN	plain GAN (10)	BigGAN-Deep (1)	BigGAN-Deep (1)
	DCGAN (37)	StyleGAN2 (20)	StyleGAN2 (20)
	LSGAN (28)	StyleGAN3 (18)	StyleGAN3 (18)
	WGAN-gp/lp (11)	VQ-GAN (8)	VQ-GAN (8)
	DRAGAN-gp/lp (24)	StyleSwin (52)	
		DDGAN (49)	
VAE	β-VAE (14)		
	DFC-VAE (16)	StyleALAE (36)	
	NVAE (43)	NVAE (43)	NVAE (43)
	VAE-BM (48)	VAE-BM (48)	
	Eff-VDVAE (12)	Eff-VDVAE (12)	Eff-VDVAE (12)
Flow	GLOW (23)	MaCow (27)	
		Residual Flow (4)	
Score		DDPM (15)	
		NCSN++ (40)	NCSN++ (40)
	RVE (22)	RVE (22)	
		LSGM (45)	
		LDM (38)	LDM (38)

From Eqn.15 and Eqn.16, we have:

$$\begin{aligned}&\exists a \in \text{FPT}(Q;P) \neq \vec{0} \;\; (\textit{i.e. } d_{\text{FPT}} \neq 0)\\ &\Rightarrow \exists \text{IPM}(Q,P) \neq 0\end{aligned} \tag{17}$$

Conversely,

$$\begin{aligned}\forall \text{IPM} : \text{IPM}(Q,P) = 0 \Rightarrow \text{FPT}(Q;P) = \{\vec{0}\}\\ (\textit{i.e. } d_{\text{FPT}}(Q;P) = 0)\end{aligned} \tag{18}$$

This means if all IPMs vanish to zero, our fingerprint also vanishes to a trivial set that only contains a zero vector.

B Related Work

Generative models and their fingerprints. Despite the recent advancement in generative models, recent works on their artifacts and biases have shown that model-generated samples contain features that can be used to identify the source models. For example, in the **color** space, the histogram of saturated and under-exposed pixels (31) and the co-occurrence matrix of color-bands (34; 35) have been shown to capture the fingerprints. In the **frequency** space, the power spectrum (6) and the decay rate of high-frequency contents (7) were able to distinguish real and GAN-generated images. More recently, **supervised learning** methods (47; 50) were used to capture fingerprints of CNN or GAN-based generators. Yet, still, there are only notions of model behaviors and their fingerprints, and a formal definition and method of computing the fingerprints are missing, which hinders a more principled study of the behaviors of generative models and the underlying factors that control their behaviors. We address this important gap in this work.

C Dataset Creation

As discussed in Sec. B, existing datasets designed for the binary discrimination of real vs. synthetic samples are not suitable for the task of model attribution (*i.e.* discriminating among multiple different generative models) in two aspects: (i) the diversity of generative models (GMs) is limited, and (ii) the variability of the models' training datasets makes the study of model fingerprint – independent of their training datasets – difficult. Rather, a proper benchmark dataset for model attribution should satisfy the following desiderata:

1. It should include GMs from various families, covering VAEs, GANs, Flows and Score-based (*i.e*. Diffusion) models
2. It should contain state-of-the-art models, in addition to the more standard models that existing works have focused on (*e.g*. ProGAN, CycleGAN, StyleGAN)
3. The generative models in the dataset should be trained on the same training set in order for the analysis on the fingerprint features to be directly attributed to the characteristics of the generative models, without confounding effects from the variability in the models' training datasets.

To this end, we designed three new datasets (GM-CelebA, GM-CHQ and GM-FFHQ; See Tbl. 2) that carefully satisfy these three desiderata. GM-CelebA contains images from generative models trained on CelebA (26), GM-CHQ from models trained on CelebA-HQ (256) (17), and GM-FFHQ from models trained on FFHQ (256) (17). To complement the existing datasets, our datasets include GMs that achieve state-of-the-art results on unconditional image synthesis, such as DDGAN(49) and StyleSwin(52) for GAN, NAVE (43) and Efficient-VDVAE (12) for VAE, and LDM (38) and LSGM (45) for diffusion models.

Fig. 6, Fig. 7 and Fig. 8 show samples from our GM256 dataset. The images are randomly sampled from each of the GMs following the process detailed in each work or codebase.

D Details on baseline fingerprinting methods

Tab. 3 summarizes baseline fingerprinting methods that we compared against our proposed definitions in Sec. 3.

Table 3: Features and datasets used in the baseline methods

Paper	Input domain	Representation	Classifiers	Metric(best)	Datasets
McCloskey18 (31)	RGB	Histogram of saturated, under-exposed pixels	SVM	AUC (0.7)	NIST MFC2018
Nataraj19 (34)	RGB	Co-occurrence matrix of pixels	CNN	EER (12.3%)	100k-Faces (StyleGAN)
Durall20 (6)	Freq.	1D power spectrum (azimuthal integral)	SVM	Binary Acc (96%)	Own (DCGAN, DRAGAN, SGAN, WGAN-gp)
Dzanic20 (7)	Freq.	Fourier spectrum (norm. by DC gain)	KNN	Binary Acc (99.2%)	Own (StyleGAN,StyleGAN2, PGGAN,VQ-VAE2,ALAE)
Wang20 (47)	Freq.	2D average spectra	CNN	LOMO, Binary Acc (84.7%)	Own (10 GANs)
Marra18 (29)	Learned	Supervised	Pretrained CNN + Finetuned (Inception-v3/XceptionNet)	LOMO^2, Binary Acc (94.49%)	Own (Real, CycleGAN per category)
Marra19 (30)	Learned	Supervised	CNN + IL	Binary Acc (99.3%)	Own (4 GANs, 1 Flow)
Yu2019 (50)	Learned	Supervised	CNN	Multi Acc (98.58%)	Own (ProGAN, SNGAN CramerGAN, MMDGAN)

E Experiment Details

E.1 Modeling choice for our fingerprints

Choice of the embedding space One main modeling decision to make when computing our fingerprints (Sec. 2.2) is the choice of the embedding space in which the true data manifold (*i.e*. natural image manifold) sit. We consider four representation spaces based on the previous works that suggest the existences of fingerprints (31; 7) and visual features (42; 51) encoded in them: RGB, Frequency, and feature spaces learned by a supervised-learning method (SL) (*e.g*. ResNet50 (13)) and by a self-supervised learning method (SSL) (*e.g*. Barlow Twins (51)). To map images to each space, we apply the following transformations (Tab. 4).

- For RGB space, we use the RGB images as is.
- For frequency space, we transform the RGB images to 2D spectrum by applying the Fast Fourier Transform (FFT) on each channel.
- For the embedding space of a supervised-learning method (SL), we use the encoder head of ResNet50 (13) pretrained on ImageNet.
- For the embedding space of a self-supervised learning method (SSL), we use the encoder head of the pretrained Barlow Twins (51).

Table 4: **Our representation spaces.** We apply the following transformations to estimate the data manifold in each embedding space.

Representation	Embedding map
RGB	Identity
FREQ	Channelwise FFT
SL	Pretrained ResNet50
SSL	Pretrained BarlowTwin

E.2 Experimental Setup

Datasets To study the effects of model architecture, learning algorithms and training data on fingerprints independently, we propose three new datasets – GM-CelebA, GM-CHQ and GM-FFHQ – each constructed from generative models trained on CelebA-64 (26), CelebA-HQ (17) and FFHQ (19), respectively. Our datasets address the absence of benchmark datasets for studying the attribution of generative models by including a variety of models from GAN, VAE, Flow and Score-based family. Tab. 2 summarizes our datasets, organized in column by the training datasets. We collect 100k images from each model. See Appendix C for details on our dataset creation process.

Baselines Existing methods of fingerprinting generative models can be categorized into three groups: color-based, frequency-based and supervised-learning. We consider key methods from each group and compare them to our proposed attribution method. See details on the baselines in Appendix D.

- Color-based: Histogram of saturated, under-exposed pixels (31), Co-occurrence matrix (34)
- Frequency-based: azimuthal-integrated power spectrum (6), high-frequency decay rate (7)
- Learned features: InceptionNet-v3 (29), XceptionNet (29), Yu19 (50), Wang20 (47)

Evaluation In the following experiments, we vary one design parameter from {model-type, training dataset, type of layers}, while fixing the other two to study the effect of that parameter independently. The varying parameter becomes the target variable in each classification task: given an input of a model-generated image, predict the varying parameter.

F Experiment Results and Discussions

F.1 Effect of model-type on fingerprints and their attributability

We first investigate how the model-type (*i.e*. a combination of model architecture and learning algorithm) affects model fingerprints and their attributability.

Metrics: To do so, we measure the accuracy of attributing artifacts of generative models, trained on the *same* training data, to the model-type of the source models (*e.g*. StyleGAN2 vs. VQ-GAN vs LDM vs ...). Since we have three datasets, each consisting of models trained on the same data (GM-CelebA, GM-CHQ, GM-FFHQ; See Tab. 2 column-wise), we evaluate the attributability on each separately. Note GM-CelebA has 13 model-types to predict, GM-CHQ 18, and GM-FFHQ 9.

Separability (FD ratio): To complement the accuracy, we measure the separability of fingerprint representations using the ratio of inter-class and intra-class Fréchet Distance (FDR) (5). The larger the ratio, the more attributable the fingerprints are to their model-type. See Appendix F.4 for the definition of FDR and how to compute it.

Results. Tab. 1 shows the results of model-type attribution in accuracy and FDR. First, the choice of model-type (BigGAN vs. StyleGAN2 vs. NVAE etc.) makes attributable artifacts in both color (McClo18, Nataraj19) and frequency (*e.g*. Durall20, Dzanic20), as well as in our artifact

Table 5: **Effect of training datasets.** **SG**: StyleGAN3 (18)

Methods	SG	NVAE	LDM
RGB(31)	0.701	0.683	0.711
Freq.(6)	0.688	0.631	0.704
Ours_{RGB}	0.622	0.612	0.645
$\text{Ours}_{\text{FREQ}}$	0.645	**0.571**	0.637
Ours_{SL}	**0.609**	0.629	**0.621**
Ours_{SSL}	0.615	0.631	0.626
Avg_{ours}	0.623	0.611	0.632

Table 6: **Layer type vs. artifacts.** Upsampling and loss type best align with the clustering of artifacts. **NL**: Non-linearity

		Layer Types (NMI ↑)				Optim.
Method	***Up***	NL	Norm	Down	Skip	***Loss***
Ours_{RGB}	0.625	0.453	0.647	0.432	0.541	0.563
$\text{Ours}_{\text{FREQ}}$	0.654	0.354	0.534	0.692	0.317	0.631
Ours_{SL}	0.613	0.452	0.481	0.546	0.434	0.677
Ours_{SSL}	0.680	0.477	0.465	0.615	0.357	0.573
Average	**0.643**	0.434	0.465	0.532	0.571	0.611

representations. This result is consistent with the exiting observations in the literature and suggests our proposed definitions capture the notion of artifacts as desired. Furthermore, our proposed methods outperforms all baselines by meaningful margins: 11.6%, 3.7%, 1.9% in each dataset, and 5.73% on average. This indicates that our methods better capture features that differentiate one generative model from another, further supporting our definitions' usefulness as fingerprints of generative models. We also note that the accuracy on GM-CelebA is higher than the accuracies on GM-CHQ and GM-FFHQ. We hypothesize this is because GM-CHQ and GM-FFHQ contain more advanced models like StyleGAN3 and NCSN++ that remove artifacts from their precedent models (18; 40): they leave less evident fingerprints, which makes the attribution more difficult. We also qualitatively compare the feature spaces using t-SNE (46) in Fig. 1. While both the original RGB and learned features (Fig. 1.(a-e)) show no clear clustering, our features (Fig. 1.(f)) show well-separated clusters.

F.2 Effect of training datasets on fingerprints and their attributability

Next, we study how the choice of training set affects the fingerprints of generative models and their attributability. To do so, we fix a generative model and vary its training sets. We evaluate the attribution of artifacts to the correct training sets. We consider three cases based on the availability of models (See Tab. 2 row-wise): (i) StyleGAN3 trained on {CelebA-HQ, FFHQ} (ii) NVAE trained on {CelebA, CelebA-HQ, FFHQ} (iii) LDM trained on {CelebA-HQ, FFHQ}.

Results. Tab. 5 shows the accuracy of attribution to the training datasets. Lower accuracy means the traces of training data are harder to be identified in the generated images. First of all, the results show that the choice of training data makes attributable traces on both color and frequency domain, as evidenced by the high accuracies on the first two rows. This suggests that existing fingerprint methods are sensitive not only to the choice of model types (as shown in Sec. F.1), but also to the choice of their training data. On the other hand, our artifact representations achieve lower accuracies, indicating they are less sensitive to the choice of training data. We argue that given our intention to capture unique features of a model, independent of is training dataset, this result is more desirable. We also note that this result is an expected outcome of the way we constructed our definition of fingerprints, as the difference between the true data manifold and the generated samples, which has the effect of "subtracting away" the fingerprints' dependence on the choice of training datasets.

F.3 Relation between artifacts of generative models and layers used in neural networks

We study the clustering of artifacts and explore the possibility of relating the clustering pattern to the types of layers. To do so, we group the layers into the following categories: Type of upsampling, non-linearity in the last layer (NL), normalization, downsampling, skip connection, and loss function. They are chosen based on previous works(6; 7; 3) that suggested them to be the causes of artifacts. We measure the clustering alignment between the clustering in a fingerprint representation and the clustering based on the type of layers using Normalized Mutual Information (NMI) (32). A higher index indicates the chosen layer parameter is more aligned with the clustering of artifacts.

Results. Table 6 reports the alignment results. Overall, we observe that upsampling and loss types best match the clustering behavior of the artifacts. First, the high NMI of the upsampling type agrees with (3; 7; 6) that suggest the upsampling layer as teh cause of high-frequency discrepancies in the model-generated images. Secondly, the high NMI for the loss type confirms the general consensus in the research community that the training objective of a generative model is one of the key factors that affect their model behaviors. Therefore, these findings experimentally confirm the general intuition

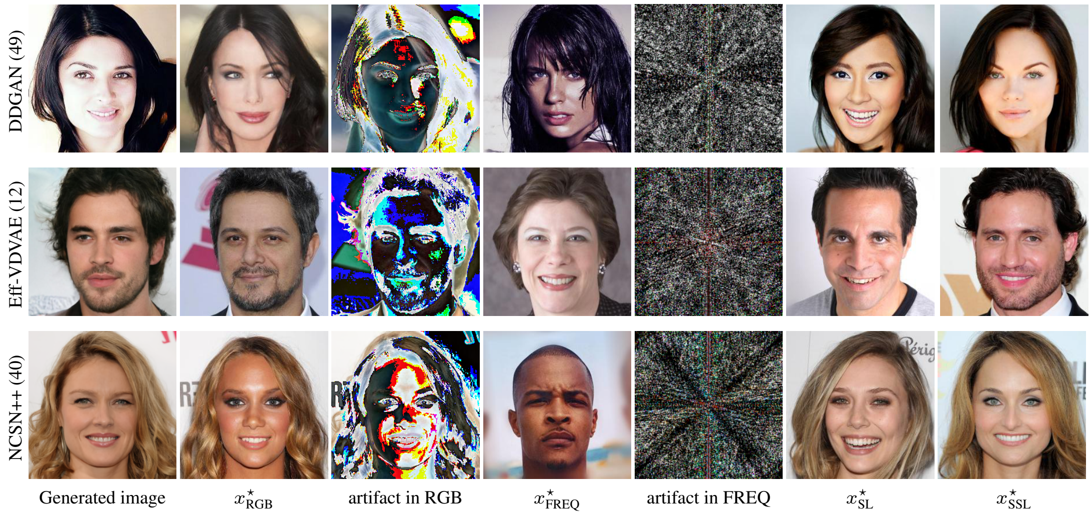


Figure 3: We visualize artifacts in generated images under our manifold-based definition (Sec. 2.1). Each row shows an original image generated by a generative model, followed by its projection to data manifolds in RGB ($x^{\star}_{\text{RGB}}$), Frequency ($x^{\star}_{\text{FREQ}}$), and learned feature spaces of SL ($x^{\star}_{\text{SL}}$) and SSL ($x^{\star}_{\text{SSL}}$). RGB and $FREQ$ correspond to the artifacts in the RGB and frequency spaces, respectively. The artifacts in SL and SSL feature spaces are not shown as they are 2048-long vectors (encoded by a pretrained ResNet50).

about the sources of limitations in generative models and also supports the utility of our definitions in studying the model behaviors.

F.4 Experiment: Feature space analysis

Metric: Fréchet Distance Ratio (FDR) We measure the separability of a fingerprint feature space using the ratio of Fréchet Distance. This measure was also used in Yu et al (50) to evaluate the learned feature space for GAN fingerprints. In our work, we use it to evaluate fingerprints in a more generalized sense in that they are to identify more diverse set of GMs (not just GANs) including many state-of-the-art models.

FDR is computed as the ratio of inter-class and intra-class Fréchet Distance (5):

$$FDR = \frac{\text{inter-class FD}}{\text{intra-class FD}} \tag{19}$$

Intra-class FD aims to capture the average tightness of a feature distribution per class, and can be measured as the FD between two disjoint sets of images in the same class. As in Yu et al (50), we split, for each class, the fingerprint features into two disjoint sets of equal size, compute their Fréchet Distance, and then average it over each class.

Inter-class FD aims to capture the average distance between feature distributions of different classes. To compute this distance, we measure the FD between two feature sets from different classes and take the average over every possible pair of (different) classes.

F.5 Visualization of artifacts of generative models

We visualize more examples of artifacts of generative models in GM-CelebA and GM-CHQ, computed under our proposed definition in Sec. 2.1. Fig. 4 shows the triplets of (generated images (x_G), its closest point to the data manifold in RGB ($x^{\star}$) and the artifact (a)). Fig. 5 visualizes the artifacts in the frequency domain from the GM-CHQ dataset.

F.6 Artifacts in RGB space (GM-CelebA)

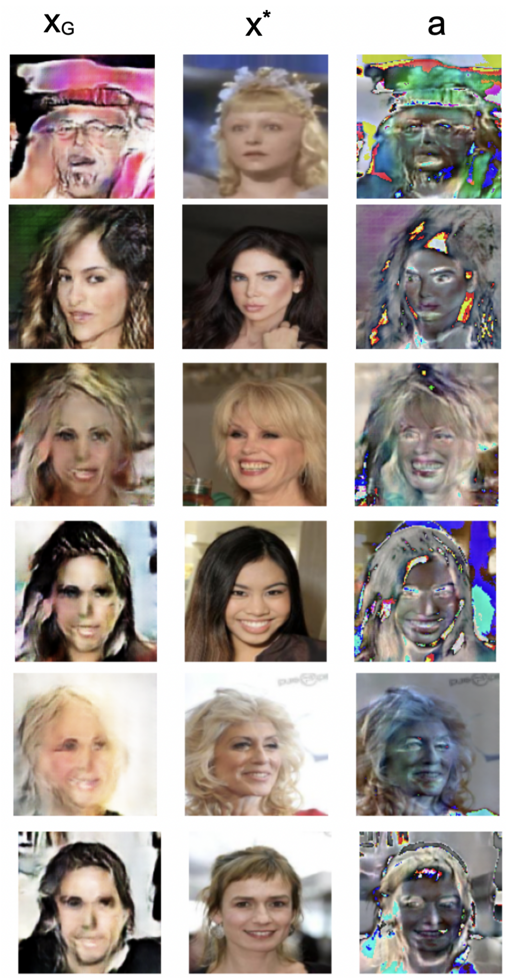


Figure 4: **Visualization of artifacts in the RGB space (GM-CelebA).** Each column corresponds to the generated images (x_G), their closest points on the data manifold ($x^\star$), and the artifacts (a). Each artifact is computed as the difference between $x^\star$ and x_G following the definition and algorithm in Sec. 2.1.

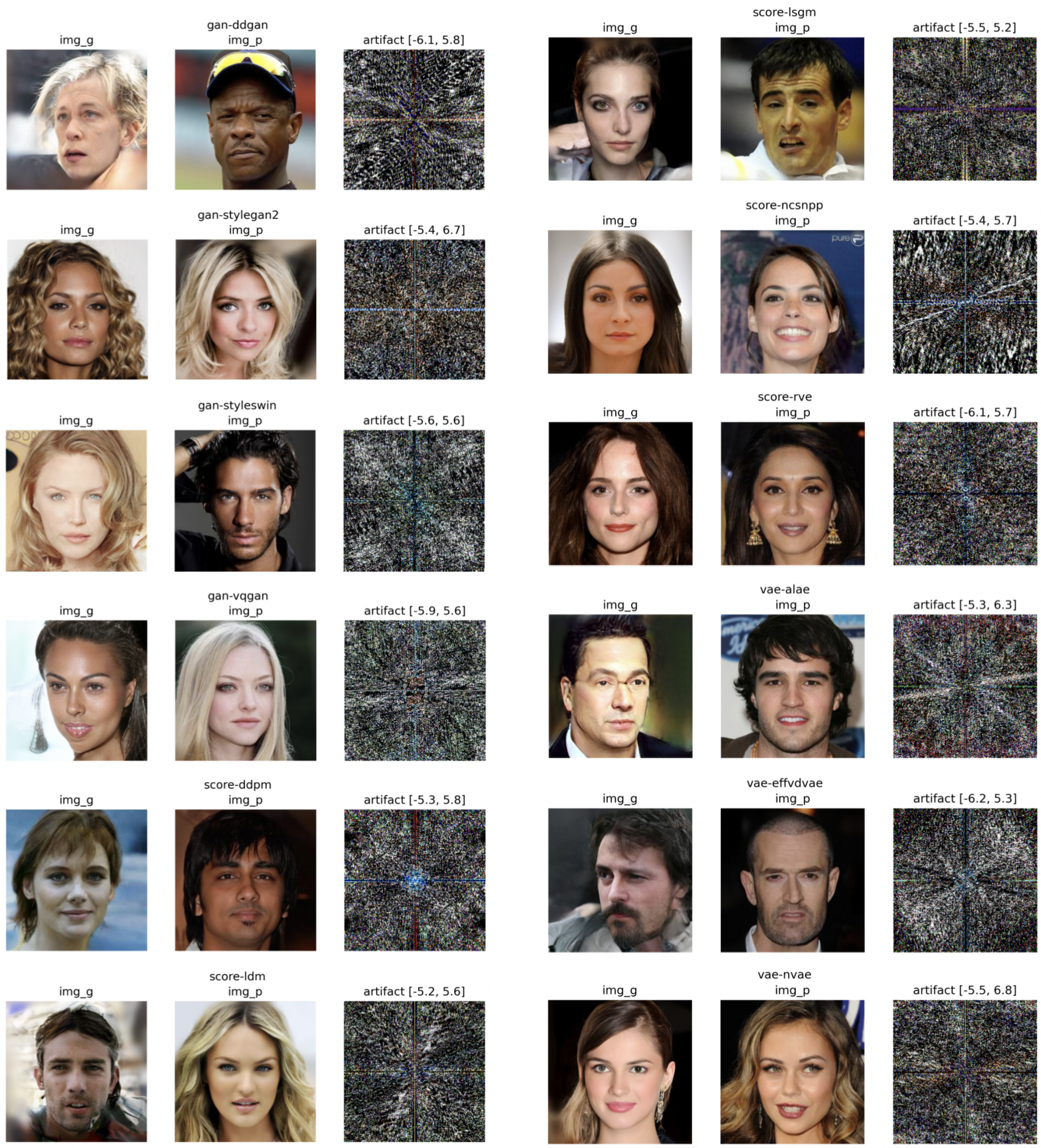


Figure 5: **Visualization of artifacts in the frequency space (GM-CHQ)**. We show some examples of triplets (model-generated image (img_g), closest point on the data manifold (img_p), artifact) from GM-CHQ dataset by computing artifacts (as defined in Sec. 2.1) in frequency domain. img_p is the point on the real data manifold that is closest to img_g in the frequency domain. Artifact is computed as the difference between the two points, img_g and img_p, after applying channelwise-FFT.

(a) DDGAN (49)

(b) StyleGAN2 (21)

(c) StyleSwin (52)

(d) VQ-GAN (8)

Figure 6: Samples from GAN models in GM-CHQ.

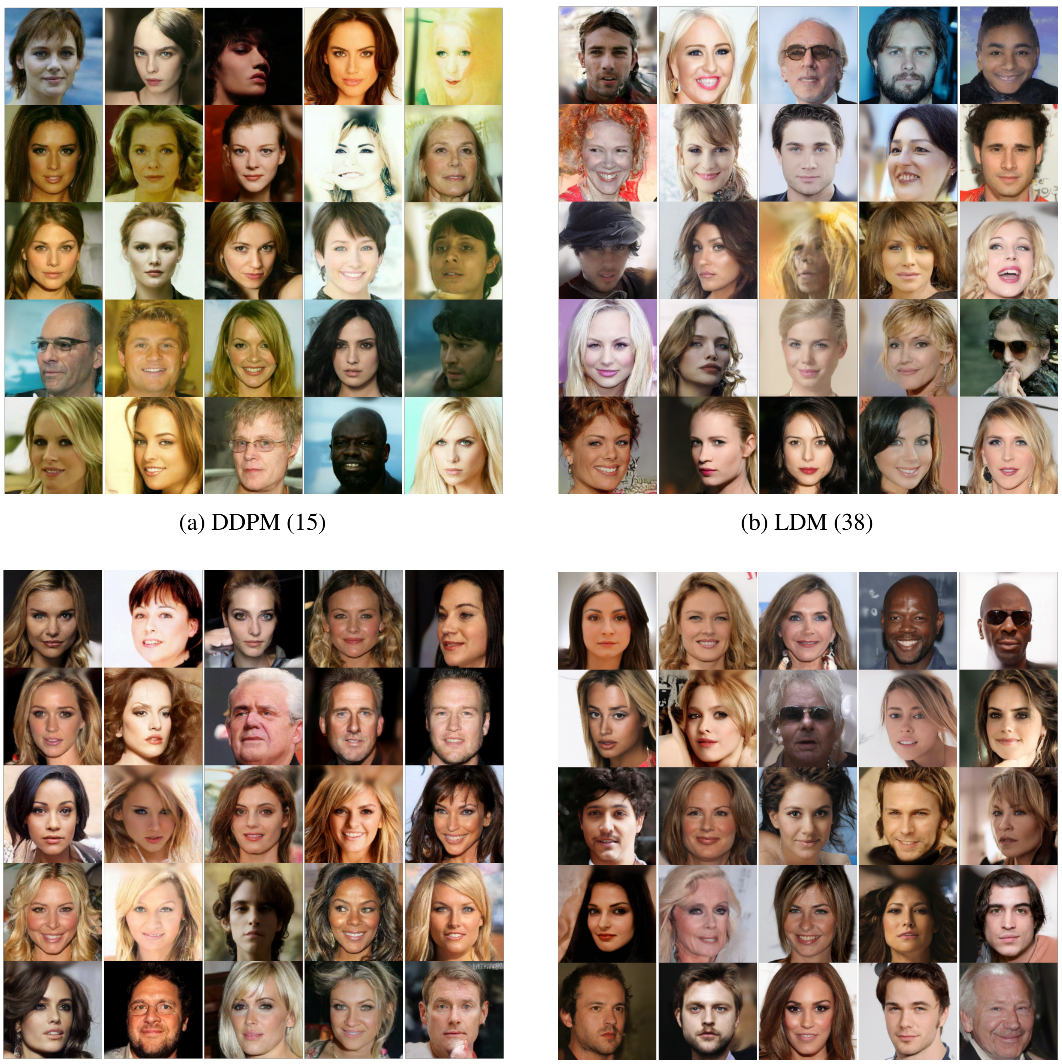
(a) StyleALAE (36)

(b) Efficient VDVAE (12)

(c) NVAE (44)

(d) VAEBM (48)

Figure 7: Samples from VAE models in GM-CHQ.



(a) DDPM (15)

(b) LDM (38)

(c) LSGM (45)

(d) NCSN++ (40)

Figure 8: Samples from score-based (a.k.a. diffusion) models in GM-CHQ.