ManiFPT: Defining and Analyzing Fingerprints of Generative Models

Hae Jin Song[1,2*] Mahyar Khayatkhoei[1,2] Wael AbdAlmageed[3]
[1]University of Southern California [2]USC Information Sciences Institute [3]Clemson University

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 extend to which these fingerprints can distinguish between various types of synthetic image and help 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 finally study its effectiveness in distinguishing a large array of different generative models. We find that using our proposed definition 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

While distinguishing synthetic images from real ones has received a lot of attention in recent years [5, 13, 27, 52, 65, 66], distinguishing between different methods of image synthesis remains mostly under-studied [59, 62]. The latter, denoted model attribution, is an interesting problem in two respects: first, in practice it is often valuable to identify the source of synthetic data – not just that it is synthetic – to differentiate between authorized vs. malicious personification [14] and digital copyright infringement [11]; second, model attribution provides a systematic way of studying the similarities and differences between various families of generative models and revealing the unique limitations of each family in learning the true distribution, thereby accelerating developments of new generative models that improve upon such limitations [3, 7, 16, 46, 59].

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

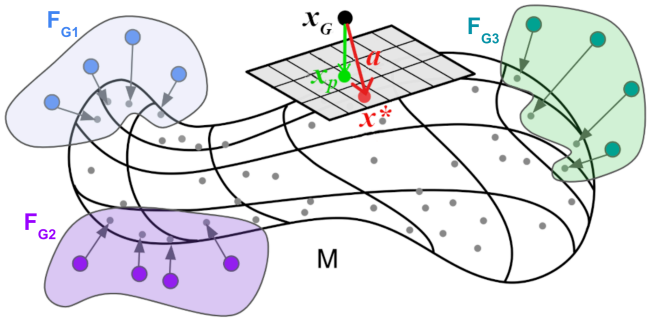
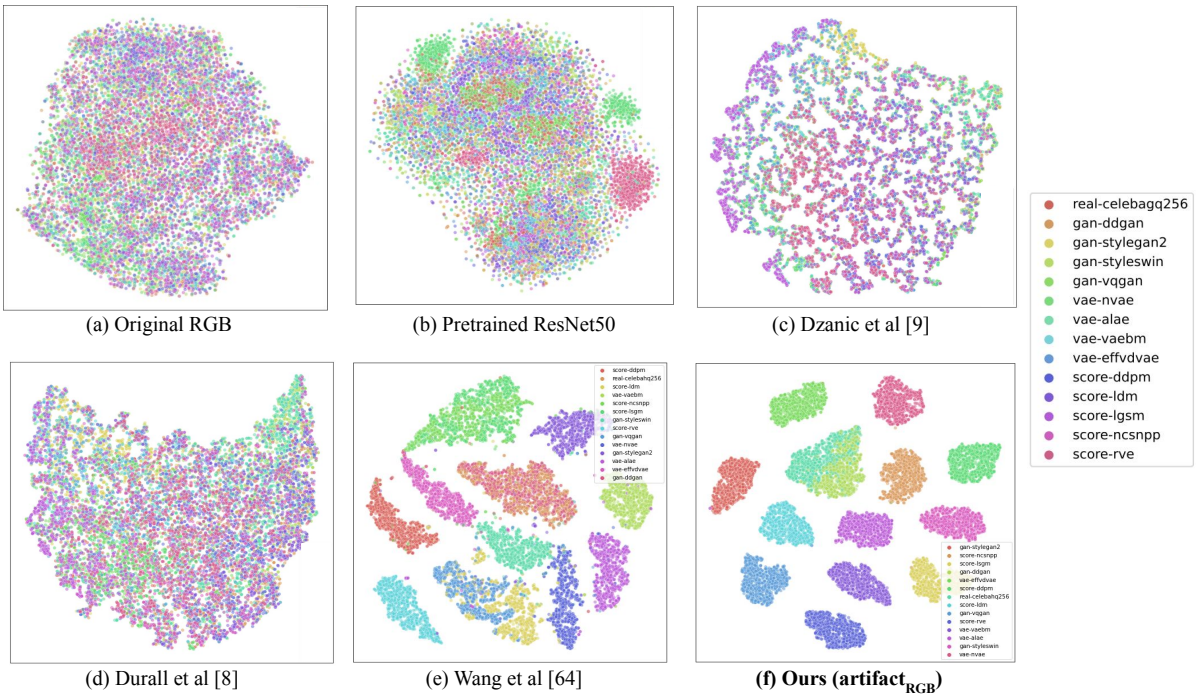


Figure 1. **Our definition of artifacts and fingerprints of a generative model.** We estimate the true data manifold $\mathcal{M}$ using real samples and compute an artifact a as the difference between a generated sample and its closest point in the real dataset. We define the fingerprint F of a generative model as the set of all its artifacts.

We study the problem of fingerprinting generative models (GMs) based on their samples with an aim to provide a formal, analytical framework to study and compare their characteristics. Recent works on GAN-generated images and DeepFake detection have observed that generative models leave unique traces of computations on their samples that are distinguishable from natural image generation process (*i.e.* image capturing and processing steps of digital cameras). Such observations include checkerboard patterns introduced by deconvolution layers in generator networks [44], spectral discrepancies due to the frequency bias of generative models [7, 37, 51, 59] and semantic inconsistencies like mismatched eye colors and asymmetric facial features [39].

The existence of such fingerprints have been supported by two streams of approaches: one by showing that a classifier can effectively distinguish samples generated by a model from real samples [59, 62], and another by directly visualizing the differences between generated and true samples [7, 8, 62]. While such works provide evidence for the existence of some fingerprints, the fingerprint itself still remains without an explicit definition. In other words, existing works define the measures of fingerprints [7, 8, 12, 39, 41], rather than fingerprints themselves. This brings about two major drawbacks in developing methods for fingerprinting



(a) Original RGB (b) Pretrained ResNet50 (c) Dzanic et al [9]

(d) Durall et al [8] (e) Wang et al [64] **(f) Ours (artifact$_{\text{RGB}}$)**

Figure 2. Features learnt 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 latent space of ResNet50 pretrained on ResNet50, (c-f) in the penultimate layer of the classifier proposed by each method trained on the task of model attribution.

GMs and studying their characteristics in a principled way. First, without a proper definition it is hard to gauge whether the different proposed measures are evaluating the same phenomenon. Secondly, and perhaps more importantly, a principled study of the fingerprints themselves – beyond showing their mere existence – is challenging.

To this end, the aim of this work is to (i) give a proper definition of generative models' fingerprints, (ii) show its sufficiency to capture the notion of fingerprints measured by existing works, and finally (iii) study properties of the fingerprints across a diverse set of GMs. We find that our proposed definition provides a useful feature space for differentiating generative models among a large array of state-of-the-art (SoTA) models, outperforms existing methods on model attribution (*i.e.* given an image, identify which generative model has generated it), and reveals interesting relations between the models.

By providing a formal definition of GM fingerprints, we address another important gap in literature, which is the lack of studies on fingerprints that are suitable for identifying different generative models in a multi-class setting: most existing methods including DeepFake detection ([37, 42, 49, 62]) focus on a binary classification of real vs. synthetic samples. Thus, whether the existing fingerprints are effective for discriminating **amongst** various generative methods has not been properly explored. To address this limitation, we introduce four new benchmark datasets (GM-CIFAR10, GM-CelebA, GM-CHQ, GM-FFHQ), each constructed from generative models trained on different training datasets (CIFAR-10 [31], CelebA-64 [33], CelebA-HQ [23] and FFHQ [25], respectively), and run extensive evaluations on them. Our contributions can be summarized as following:

- We formalize the definition of fingerprints in generative models, and propose a practical algorithm to compute them from finite samples.
- We provide theoretical justification of our definition by relating it to two prominent metrics for distinguishing generative models: Precision and Recall (P&R) [32, 50] and integral probability metrics (IPMs) [40, 54].
- 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 the exiting works.
- We use the proposed definitions to study the effects of design factors (*e.g.* type of layers) on the model fingerprints, and observe that the choice of loss functions and upsampling have the most significant effect. Our findings confirm the general intuition in the research community about the sources of limitations in generative models [7, 8], thereby showing the utility of our definitions.

2. Related Work

Generative models and their fingerprints. Despite the advancement in generative models, recent works on their artifacts and biases have shown that the samples they generate contain features that can be used to identify the source models. The existing observations of such features are categorized into four kinds: steganalysis-based, color-based, frequency-based and learning-based features.

Steganalysis-based features. Inspired by the methods of fingerprinting digital cameras based on the PRNU patterns (residual patterns), Marra et al. [37] investigates whether GAN image synthesis models leave unique and stable marks on the images they generate and whether they can be used to address image attribution task. They propose the residual image as a representation of the image-level fingerprint, and estimates the model-level fingerprint as the average of the residual images by the source generative model. We share the same goal as their work in that we want to learn unique, discriminative representations of generative models based on their samples (*i.e*. fingerprinting GMs), yet expand the set of GMs under consideration to more diverse and more recent models such as the state-of-the-art GANs (DDGAN) and score-based models like DDPM, NCSN++ and LSGM.

Color-based features. McCloskey et al. [39] shows that the histogram of saturated and under-exposed pixels in GAN images provides a sufficient signal to discriminate StyleGAN-generated images from the real images. Nataraj et al. [41] and Nowroozi et al. [43] use co-occurrence matrix on color-bands as the input representation of image fingerprints and feed them to CNN to predict Real vs. GAN images.

Frequency-based features. Durall et al. [7] shows that current generative models fail to correctly reproduce the spectral distributions of the images in their training dataset by analyzing the spectral statistics of real and synthetic images. They propose a hand-crafted, 1-dimensional feature as frequency-based fingerprint feature, computed via azimuthal integration over the 2D spectrum of each image. They show that this frequency feature is sufficient to distinguish real and synthetic images on the dataset containing AutoEncoders, deepfake manipulation methods and four GANs trained on CelebA (DCGAN, DRAGAN, LSGAN, WGAN-GP). Dzanic et al. [8] shows that synthetic images manifest different spectral statistics at high-frequencies in two aspects – the amount of high-frequency contents and the decay rate – and proposes two parameters that capture these two aspects from a reduced spectru as fingerprints. They consider three GANs (StyleGAN1, StyleGAN2, ProGAN) and two VAEs (VQ-VAE2, ALAE). However, their work is limited to the binary classification and do not address model attribution among different generative models. Wang et al. [59] hypothesizes that CNN-based generators leave common fingerprints on the images they generates which can be used to distinguish them from natural images. This feature can be considered an image-level fingerprint, *i.e*. features on individual images, yet their work does not make an explicit definition of model-level fingerprints. Our work considers both image-level and model-level fingerprints, and further clarifies the two notions by formally defining them in Sec. 3.1 as "GM artifacts" (for image-level features) and "GM fingerprints" (for model-level features).

Learning-based features. Marra et al. [36] uses CNN classifier (pretrained and then fine-tuned on their dataset) to predict the real vs. fake images. Yu et al. [62] is one of the first works to differentiate the notion of "image fingerprints" and "model fingerprints". In their work, "image fingerprints" refers to features that exist across sample by the same model, and "model fingerprints" refers to features that distinguish one source model from another (e.g., fingerprint of progan-seed0 vs. sngan-seed10). However, the formal definition of such fingerprints were not provided in their work.

3. ManiFPT: Manifold-based Fingerprints of generative models

As discussed in Sec. 1 and Sec. 2, despite existing works that provide evidence for the existence of fingerprints and artifacts in generative models, the concrete definitions of these terms remain unclear. In this section, we motivate and propose formal definitions of artifacts and fingerprints of generative models, and then describe a practical algorithm for computing them from observed samples.

3.1. Definitions of GM artifacts and fingerprints

Intuitively, the artifacts and fingerprints of generative models are the traces of their imperfection in modeling the generative process of real data, left consistently on the samples they generate. From the point of view of manifold learning [2, 10], which hypothesize that many high-dimensional real-world datasets (*e.g*. images and videos) lie on a lower-dimensional manifold, such *imperfections* of generative models can be formalized in terms of the *deviation of generated samples from the true data manifold*. More concretely, consider a generative model G which is trained on a dataset X of real samples that lie on a data manifold $\mathcal{M}$, with P_G the induced probability distribution of G, S_G its support, and x_G a sample generated by G:

Definition 3.1 (Artifact)**.** An artifact $a_{\mathcal{M}}$ left by generative model G on a sample x_G generated by G is defined as the difference between x_G and closest point x^* on the manifold $\mathcal{M}$ of the dataset used to train G, equipped with a distance metric d_M:

$$x^* := \underset{x \in M}{\operatorname{argmin}}\, d_M(x_G, x) \tag{1}$$

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

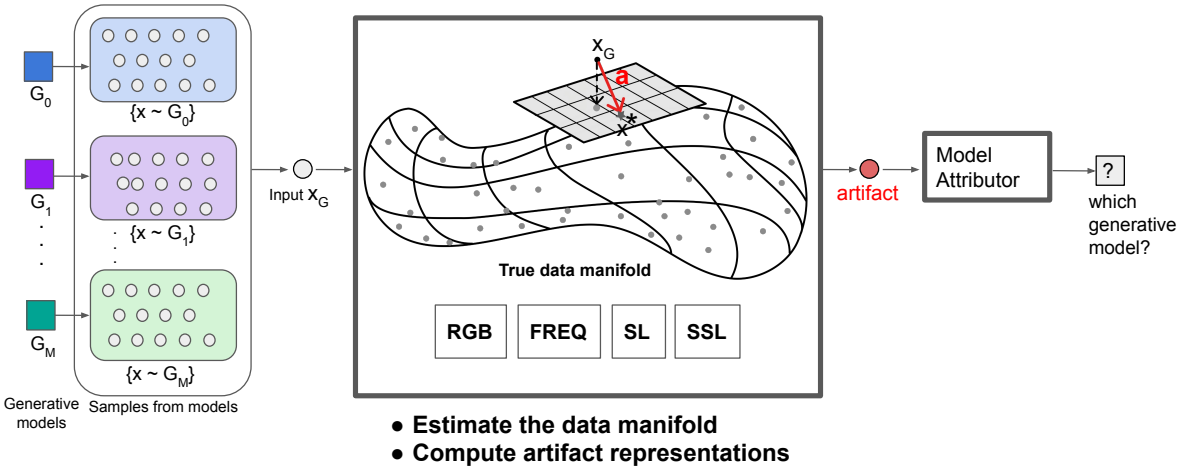


Figure 3. **Our attribution method.** We propose a model attribution method based on our definition of artifact as deviations from an estimate data manifold. Given input images X_G, we first map the images to a chosen embedding space (RGB, Frequency, a feature space of a pretrained supervised-learning (SL) or self-supervised leanring (SSL) network) and compute their artifacts a. We then pass the artifacts to a ResNet50-based attribution network (Model Attributor) and fine-tune the network to identify the source generative model under the (multi-class) cross-entropy loss.

Definition 3.2 (Fingerprint)**.** The fingerprint of a generative model G, whose support is S_G, with respect to the data manifold $\mathcal{M}$ is defined as the set of all its artifacts:

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

Figure 1 illustrates our proposed definitions.

3.2. Estimation of GM artifacts and fingerprints

Estimating the data manifold. Since we do not have an access to the data manifold $\mathcal{M}$ on which the real image lie (*i.e.* the natural image manifold), we need to estimate it using the observed samples at hand: To this end, 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. We experiment with four possibilities (RGB, Frequency, and learned spaces of a supervised-learning method and a self-supervised learning method) in Sec. 4.1.

Computing the artifact of a model-generated sample. An artifact of a 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. We use the Euclidean distance, *i.e.* $d_M(x, x_G) := ||x - x_G||^2$
2. Compute the artifact as difference, $a(x_G) = x_G - x^*$

Fig. 4 shows some examples of the projections and artifacts in RGB, FREQ, SL and SSL spaces computed in this way.

Fingerprint of a model Given a set of model generated samples $X_G = \{x_i\}_{i=1}^N$ where $x_i \sim P_G$, we estimate its fingerprint by computing an artifact of each sample in X_G, *i.e.* $F_G = \{a(x) | x \in X_G\}$.

3.3. Theoretical justification of our definitions

Our proposed definitions of GM artifacts and fingerprints are closely related to two prominent metrics for distinguishing generative models: Precision and Recall (P&R) [32, 50] and integral probability metrics (IPMs) [40, 54]. The most fundamental relation is that under our definition, the fingerprint is a non-zero set 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 3.1. Let $d_{\text{FPT}}(Q, P)$ be the norm of a largest artifact vector in $\text{FPT}(Q, P)$ defined as,

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

$d_{\text{FPT}}(Q, P)$ 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:

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

By the definition of P&R in Defn (2) of [32],

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

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,

$$\text{FPT}(P, Q) = \{\vec{0}\} \Leftrightarrow d_{\text{FPT}}(P, Q) = 0 \quad (10)$$

$$\Leftrightarrow \text{Recall}(Q, P) = 1 \quad (11)$$

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:

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

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

Additionally, we have the property of equal supports:

$$\text{FPT}(Q, P) = \{\vec{0}\} \text{ and } \text{FPT}(P, Q) = \{\vec{0}\}$$

$$\Leftrightarrow \quad \text{Supp(P)} = \text{Supp}(Q) \quad \textbf{(equal supports)} \quad (14)$$

The property of equal supports implies the degree of our fingerprint's expressivity in regards to the difference between P and Q: as long as there is at least one generated sample not 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) [40, 54], which include MMD and Wasserstein distance, in the following way: By the property of equal supports in Eqn. 14,

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

By the definition of IPMs [40, 54],

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

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

$$\exists a \in \text{FPT}(Q, P) \neq \vec{0} \;\; (i.e.\; d_{\text{FPT}} \neq 0)$$

$$\Rightarrow \exists \text{IPM}(Q, P) \neq 0 \quad (17)$$

Conversely,

$$\forall \text{IPM} : \text{IPM}(Q, P) = 0 \Rightarrow \text{FPT}(Q, P) = \{\vec{0}\}$$

$$(i.e.\; d_{\text{FPT}}(Q, P) = 0) \quad (18)$$

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

Family	GM-CIFAR10	GM-CelebA	GM-CHQ	GM-FFHQ
Real	CIFAR-10 [31]	CelebA [33]	CelebA-HQ (256) [23]	FFHQ (256) [25]
GAN	BigGAN-Deep [1]	plain GAN [15]	BigGAN-Deep [1]	BigGAN-Deep [1]
	StyleGAN2 [26]	DCGAN [47]	StyleGAN2 [26]	StyleGAN2 [26]
		LSGAN [35]	StyleGAN3 [24]	StyleGAN3 [24]
	WGAN-gp [17]	WGAN-gp/lp [17]	VQ-GAN [9]	VQ-GAN [9]
		DRAGAN-gp/lp [30]	StyleSwin [64]	
	DDGAN [61]		DDGAN [61]	
VAE		β-VAE [20]		
		DFC-VAE [22]	StyleALAE [45]	
	NVAE [56]	NVAE [56]	NVAE [56]	NVAE [56]
	VAE-BM [60]	VAE-BM [60]	VAE-BM [60]	
	Eff-VDVAE [18]	Eff-VDVAE [18]	Eff-VDVAE [18]	Eff-VDVAE [18]
Flow	GLOW [29]	GLOW [29]		
	MaCow [34]		MaCow [34]	
			Residual Flow [4]	
Score	DDPM [21]	DDPM [21]	DDPM [21]	
			NCSN++ [53]	NCSN++ [53]
	RVE [28]	RVE [28]	RVE [28]	
	LSGM [57]		LSGM [57]	
			LDM [48]	LDM [48]

Table 1. **Our experimental dataset of generation models.** We collect images from a diverse set of generative models trained on four different datasets (CIFAR10, CelebA, CelebA-HQ(256), FFHQ(256)) and study model fingerprints and their attributability. **Real**: training datasets of the generative models. **Score**: score-based (aka. diffusion) models.

3.4. Attribution network

The goal of our attribution network is to predict the source generative model of an observed image. It takes as input our artifact representation of an image (as defined in 3.1) and predicts the identity of its source generative model. We use ResNet50 [19] as our attribution network and finetune it with the standard cross-entropy loss. The network is trained for source model attribution over images generated by different generative models. The first step in our attribution procedure is to represent the input image as an artifact feature by computing its deviation from an estimate data manifold as discussed in Sec. 3.2. Note that the main novelty in our attribution method is in representing each input as an artifact, *i.e.* a deviation from the estimated data manifold, where we consider either RGB, frequency, or ResNet-learned feature space as the embedding space of the manifold (Sec 3.2). Figure 3 illustrates our attribution process.

4. Experiments

We start our experiments by hypothesizing the existence of fingerprints for individual generative models. Sec. 4.1 explains our experimental setup. Sec. 4.2 and Sec. 4.3 explore the existence of GM fingerprints and the attributability of generative models via classification and feature space analysis. Building on experimental supports for their existence, Sec. 4.5 studies the clustering structure of the set of GM artifacts, and their relation to different generative models.

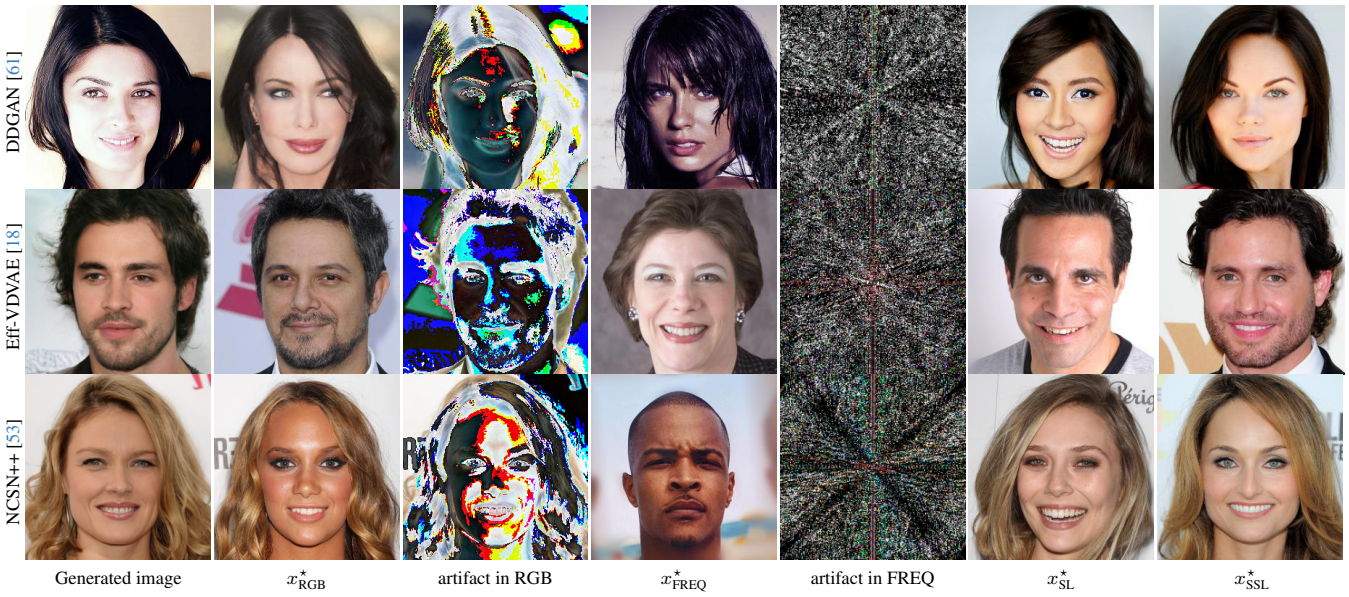


Figure 4. We visualize artifacts in generated images under our manifold-based definition (Sec. 3.1). Each row shows an original image generated by a generative model, followed by its projection to data manifolds in RGB ($x^\star_{\text{RGB}}$), Frequency ($x^\star_{\text{FREQ}}$), and learned feature spaces of SL ($x^\star_{\text{SL}}$) and SSL ($x^\star_{\text{SSL}}$). The third and fourth columns show our definition of artifacts in the RGB and frequency spaces, respectively. Note that artifacts in SL and SSL spaces are not shown as they are 2048-long vectors (in the embedding space of a pretrained ResNet50).

4.1. Experimental setup

Datasets To evaluate how well different fingerprints perform on model attribution in a truly multi-class setting, we propose four new datasets – GM-CIFAR10, GM-CelebA, GM-CHQ and GM-FFHQ – constructed from real datasets and generative models trained on CIFAR-10 [31], CelebA-64 [33], CelebA-HQ(256) [23] and FFHQ(256) [25], respectively. We collect 100k images from each generative model. 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, and state-of-the-art generative models (*e.g.* DDGAN, VAE-BM, LSGM) that have not been considered before. Tab. 1 summarizes our datasets, organized in column by the training datasets. See Appendix **??** for details on our dataset creation process. We highlight that each dataset exclusively consists of models trained on the same training dataset: this consistency is important for studying the effects of model architectures and datasets on attribution independently, which we study in Sec. 4.2 and Sec. 4.4.

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 method. See details on the baselines in Appendix **??**.

- Color-based: Histogram of saturated, under-exposed pixels [39], Co-occurrence matrix [41]
- Frequency-based: 1-dim power spectrum via azimuthal integration on DCT [7], high-frequency decay parameters fitted to normalized reduced spectra [8]
- Features via supervised-learning: InceptionNet-v3 [36], XceptionNet [36], Yu19 [62], Wang20 [59]

Choice of the embedding space One main modeling decision to make when computing our fingerprints (Sec. 3.2) 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 [8, 39] and visual features [55, 63] encoded in them: RGB, Frequency, and feature spaces learned by a supervised-learning method (SL) (*e.g.* ResNet50 [19]) and by a self-supervised learning method (SSL) (*e.g.* Barlow Twins [63]). To map images to each space, we apply the following transformations (Tab. 3).

- 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 [19] pretrained on ImageNet.
- For the embedding space of a self-supervised learning method (SSL), we use the encoder head of the pretrained Barlow Twins [63].

4.2. Existence of fingerprints in generative models

We test the existence of fingerprints on the generated samples by training a classifier for model attribution. A high test accuracy implies the classifier is able to extract features from

Methods	GM-CIFAR10		GM-CelebA		GM-CHQ		GM-FFHQ	
	Acc.(%)↑	FDR↑	Acc.(%)↑	FDR↑	Acc.(%)↑	FDR↑	Acc.(%)↑	FDR↑
McClo18 [39]	40.223 ± 1.10	32.4	62.6 ± 2.314	70.2	57.4 ± 2.013	36.3	50.8 ± 0.341	26.3
Nataraj19 [41]	46.291 ± 1.43	36.7	61.1 ± 2.203	74.0	56.3 ± 1.325	37.9	51.3 ± 0.581	35.3
Durall20 [7]	57.293 ± 0.93	46.5	62.2 ± 2.243	75.5	59.1 ± 1.301	38.8	60.9 ± 0.255	37.9
Dzanic20 [8]	56.123 ± 1.21	43.1	61.6 ± 2.029	88.1	56.9 ± 1.215	38.2	55.7 ± 0.324	30.3
Wang20 [59]	62.23 ± 0.84	53.6	62.2 ± 1.203	89.8	59.5 ± 1.252	30.3	64.2 ± 0.310	37.9
Marra18 [36]	55.944 ± 1.09	41.2	63.1 ± 1.103	83.4	51.3 ± 1.281	20.5	53.2 ± 0.218	30.4
Marra19 [38]	60.71 ± 1.24	47.2	61.1 ± 1.729	101.4	59.1 ± 1.27	34.9	51.8 ± 0.233	30.9
Yu19 [62]	62.01 ± 0.79	50.1	60.6 ± 1.103	111.4	61.1 ± 1.122	74.5	60.5 ± 0.105	35.1
ManiFPT$_{\text{RGB}}$	69.48 ± 1.08	55.2	70.5 ± 1.565	115.3	63.7 ± 1.238	64.2	65.3 ± 0.125	50.1
ManiFPT$_{\text{FREQ}}$	70.191 ± 0.96	57.2	72.8 ± 1.321	120.9	**64.8** ± 1.124	70.1	**66.1** ± 0.207	**57.6**
ManiFPT$_{\text{SL}}$	**72.018** ± 0.92	**58.9**	73.6 ± 1.102	**168.0**	62.3 ± 1.221	**77.2**	63.2 ± 0.305	49.8
ManiFPT$_{\text{SSL}}$	70.177 ± 1.13	56.1	**74.7** ± 1.121	125.9	61.9 ± 1.351	63.3	63.8 ± 0.203	40.9

Table 2. **Model attribution results.** We evaluate different artifact features on the task of predicting the source generative model of a generated sample. Separability of the feature spaces are measured in FD ratio (FDR). Higher FDR means better separability. Our methods (*ART*'s) based on the proposed definition of artifacts outperform all baseline methods on four different datasets.

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

Table 3. **Our embedding spaces.** To estimate the data manifold in a suitable embedding space, we apply each transformation to input images.

the images that are distinct signatures of their source models, thereby supporting the existence of their fingerprints.

Metrics. We evaluate the attributability of baseline methods listed in Tab. **??** and our proposed methods on our GM datasets. The performance is measured in accuracy(%). We consider four variants of our attribution method by representing the artifacts in different embedding spaces ({RGB, Frequency, Supervised-learning (SL) and self-supervised learning (SSL)} spaces; Sec. 3.1). They are referred to as *ManiFPT*$_{\text{RGB}}$, *ManiFPT*$_{\text{FREQ}}$, *ManiFPT*$_{\text{SL}}$, and *ManiFPT*$_{\text{SSL}}$ in Tab. 2.

Evaluation Protocol. Each dataset consists of real images and generated images from M generative models, and each image is labelled with the identity of their source model, *e.g.* 0 for Real, 1 for G_1, ..., M for G_M. We split the data into train, val, test in ratio of $7:2:1$, train the classifier on the train split with the cross-entropy loss over the labels, and measure the accuracy on the test split.

Results. Tab. 2 shows the result of model attribution using each fingerprinting method. First of all, we notice that the overall accuracy is lower on GM-CHQ, than on GM-CelebA. This aligns with our intuition that attributing samples becomes harder when they are generated by more advanced models that match the true data-generating process better. Another possible reason for this phenomenon is that SoTA models in GM-CHQ are often hybrid, meaning a model (*e.g.* DDGAN) incorporates architectural and optimization techniques from multiple families of GMs (*e.g.* a combination of adversarial training and diffusion sampling), thereby making the artifacts the models generate also become a mixture, and harder to attribute to a single model instance. Secondly, we observe that the fingerprints based on hand-crafted features such as the histogram of saturated, under-exposed pixels [39], co-occurrence matrix of RGB pixels [41] and 1D power spectrum [7] perform worse than the fingerprints learned with CNN-based classifiers([36], [62]). Lastly, our attribution method outperforms all the existing methods on all datasets by meaningful margins (11.6%, 3.7%, 1.9% in each dataset, and 5.73% on average), thus supporting our definitions' usefulness as fingerprints of generative models.

4.3. Feature space analysis

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) [6]. The larger the ratio, the more attributable the fingerprints are to their model-type. See Appendix **??** for the definition of FDR and how to compute it. Tab. 2 shows the FD ratios computed for the fingerprint representations on the test datasets. The FDRs are significantly higher for learned representations (Row of Wang20 and below) than color-based (McClo8, Nataraj19) or frequency-based fingerprints (Durall20, Dzanic20). In particular, our artifact-based feature spaces achieve improved FDRs, in alignment with the attribution results in classification accuracy.

Methods	C10→CA	CA→C10	CHQ→FFHQ	FFHQ→CHQ
McClo18 [39]	52.3	43.2	34.2	31.2
Nataraj19 [41]	56.2	46.1	42.1	40.4
Durall20 [7]	60.1	53.5	51.9	42.6
Dzanic20 [8]	56.9	54.7	45.2	42.5
Wang20 [59]	62.5	60.1	61.4	53.4
Marra18 [36]	57.0	58.4	50.2	35.9
Marra19 [38]	61.0	58.6	54.3	30.3
Yu19 [62]	60.5	60.4	55.2	50.3
ManiFPT$_{\text{RGB}}$	62.7	60.2	**66.3**	53.2
ManiFPT$_{\text{FREQ}}$	65.8	62.1	63.5	54.3
ManiFPT$_{\text{SL}}$	**67.7**	60.3	57.6	**56.9**
ManiFPT$_{\text{SSL}}$	66.3	**63.0**	58.1	53.5

Table 4. **Generalization of model attribution across datasets.** We evaluate how well baselines and our fingerprints generalize across training datasets. We consider two scenarios: (i) generalization across GM-CIFAR10 and GM-CelebA, and (ii) generalization across GM-CHQ and GM-FFHQ. For each case, we train attribution methods on one set of generative models (*e.g.* GM-CIFAR10) and test on a different set of models (*e.g.* GM-CelebA). Our artifact-based attribution method outperforms all baseline methods in both scenarios. **C10**: CIFAR-10. **CA**: CelebA. **CHQ**: CelebA-HQ.

tSNE of fingerprint features. We qualitatively compare the feature spaces of six different fingerprint representations using t-SNE [58] in Fig. 2. While both the original RGB features or features extracted by a pretrained ResNet50 [19] show no clear clustering, the features learned using our artifact representations (Fig. 2.f) show much better separated clusters. Note that each cluster correctly corresponds to its source generative model, which supports the utility of our features as fingerprints of the generative models.

4.4. Cross-dataset generalization

We study the generalizability of attribution methods across datasets. This generalizability is important in practice, when, *e.g.* a correct attribution is needed even when different end users train new models using their own datasets. We consider two scenarios: generalization (i) across GM-CIFAR10 and GM-CelebA, and (ii) across GM-CHQ and GM-FFHQ.

Evaluation. In each scenario, we train attribution methods on the training dataset (*e.g.* GM-CIFAR10) and test their accuracies on an unseen dataset (*e.g.* GM-CelebA).

Results. Tab. 4 shows the result of the accuracies of the cross-dataset generalization for GM-CIFAR10 $\leftrightarrow$ GM-CelebA and for GM-CHQ $\leftrightarrow$ GM-FFHQ. First of all, our manifold-base attribution methods outperform existing methods in both scenarios. Note that CIFAR-10 and CelebA contain images from different domains (CIFAR-10: objects and animals vs. CelebA: human faces), while CelebA-HQ and FFHQ both contains facial images. Therefore, the high accuracies in both scenarios indicate that our method generalize not only across datasets of similar semantics (CHQ $\leftrightarrow$ FFHQ), but also across of different semantics (CIFAR-10 $\leftrightarrow$ CelebA). Overall, these higher accuracies show that our methods are more robust to the change of training datasets of the generative models, supporting the efficacy of our methods in practice where attribution is needed amidst various end users who can train new models using their own datasets. This result is in alignment with the way we constructed our definition of fingerprints: since they are defined as the differences between the true data manifold and the generated samples, it has the effect of "subtracting away" the fingerprints' dependence on the choice of training datasets, thereby improving the cross-dataset generalizability.

	(NMI↑)	Layer	Types			Optim.
Method	***Up***	NL	Norm	Down	Skip	***Loss***
ManiFPT$_{\text{RGB}}$	0.625	0.453	0.647	0.432	0.541	0.563
ManiFPT$_{\text{FREQ}}$	0.654	0.354	0.534	0.692	0.317	0.631
ManiFPT$_{\text{SL}}$	0.613	0.452	0.481	0.546	0.434	0.677
ManiFPT$_{\text{SSL}}$	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

Table 5. **Clustering structure of fingerprints.** Types of upsampling and loss best align with the clustering. **NL**: Non-linearity

4.5. Clustering structure of GM fingerprints

Lastly, we study the structure of GM fingerprints by studying the alignment of their clustering pattern to the hyperparameters used in the design of generative models (*e.g.* type of sampling layers and loss functions). Table 5 reports the alignment results. Overall, we observe that upsampling and loss types best match the clustering behavior of the artifacts, experimentally confirming the general intuition about the sources of limitations in generative models and supporting the utility of our definitions in studying the model behaviors. See full discussion in Appendix 4.5.

5. Conclusion

Our work addresses the emerging problem of differentiating generative models and attributing generated samples to their proper source models. We study this problem in a principled way by proposing formal definitions of artifacts and fingerprints of generative models, which has been missing in the literature. We further provide a theoretical justification of our proposed definitions in relation to two key metrics on generative models (Precision and Recall, and IPMs), and demonstrate their practical usefulness in differentiating a large array of state-of-the-art models, of various types (GAN, VAE, Flow and Score-based). Our method outperforms existing methods on model attribution, generalizes better across datasets, and learns a feature space effective for differentiating generative models. We believe our definitions will lay an important step towards formalizing the characteristics of generative models and prepare for their integration into our society by helping to develop more effective attribution methods.

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. 5

[2] Lawrence Cayton. Algorithms for manifold learning. *Univ. of California at San Diego Tech. Rep*, 12(1-17):1, 2005. 3

[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]. 1

[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

[5] Davide Cozzolino, Justus Thies, A. Rössler, C. Riess, M. Nießner, and L. Verdoliva. ForensicTransfer: Weakly-supervised Domain Adaptation for Forgery Detection. *ArXiv*, 2018. 1

[6] D.C. Dowson and B.V. Landau. The fréchet distance between multivariate normal distributions. *Journal of multivariate analysis*, 12(3):450–455, 1982. 7

[7] 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. 1, 2, 3, 6, 7, 8

[8] 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. 1, 2, 3, 6, 7, 8

[9] 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. 5

[10] Charles Fefferman, Sanjoy Mitter, and Hariharan Narayanan. Testing the manifold hypothesis. *Journal of the American Mathematical Society*, 29(4):983–1049, 2016. 3

[11] Giorgio Franceschelli and Mirco Musolesi. Copyright in generative deep learning. *Data & Policy*, 4:e17, 2022. Publisher: Cambridge University Press. 1

[12] Joel Cameron Frank, Thorsten Eisenhofer, Lea Schönherr, Asja Fischer, Dorothea Kolossa, and Thorsten Holz. Leveraging frequency analysis for deep fake image recognition. *ArXiv*, abs/2003.08685, 2020. 1

[13] Sheldon Fung, Xuequan Lu, Chao Zhang, and Chang-Tsun Li. DeepfakeUCL: Deepfake Detection via Unsupervised Contrastive Learning. *2021 International Joint Conference on Neural Networks (IJCNN)*, 2021. 1

[14] Oliver Giudice, Luca Guarnera, and Sebastiano Battiato. Fighting Deepfakes by Detecting GAN DCT Anomalies. *Journal of Imaging*, 7(8):128, July 2021. 1

[15] 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. 5

[16] Luca Guarnera, Oliver Giudice, and Sebastiano Battiato. DeepFake Detection by Analyzing Convolutional Traces. Apr. 2020. 1

[17] Ishaan Gulrajani, Faruk Ahmed, Martín Arjovsky, Vincent Dumoulin, and Aaron C. Courville. Improved training of wasserstein gans. In *NIPS*, 2017. 5

[18] Louay Hazami, Rayhane Mama, and Ragavan Thurairatnam. Efficient-VDVAE: Less is more, Apr. 2022. 5, 6

[19] 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. 5, 6, 8

[20] 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. 5

[21] 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. 5

[22] 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. 5

[23] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of gans for improved quality, stability, and variation. *ArXiv*, abs/1710.10196, 2018. 2, 5, 6

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

[25] 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. 2, 5, 6

[26] 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. 5

[27] Aminollah Khormali and Jiann-Shiun Yuan. DFDT: An End-to-End DeepFake Detection Framework Using Vision Transformer. *Applied Sciences*, 2022. 1

[28] 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. 5

[29] 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. 5

[30] Naveen Kodali, Jacob Abernethy, James Hays, and Zsolt Kira. On convergence and stability of gans. *arXiv preprint arXiv:1705.07215*, 2017. 5

[31] Alex Krizhevsky. Learning multiple layers of features from tiny images. 2009. 2, 5, 6

[32] 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. 2, 4, 5

[33] 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. 2, 5, 6

[34] Xuezhe Ma and Eduard H. Hovy. Macow: Masked convolutional generative flow. In *NeurIPS*, 2019. 5

[35] 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. 5

[36] 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. 3, 6, 7, 8

[37] Francesco Marra, Diego Gragnaniello, Luisa Verdoliva, and Giovanni Poggi. Do gans leave artificial fingerprints? In *2019 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR)*, pages 506–511. IEEE, 2019. 1, 2, 3

[38] 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. 7, 8

[39] Scott McCloskey and Michael Albright. Detecting GAN-generated Imagery using Color Cues, Dec. 2018. 1, 3, 6, 7, 8

[40] Alfred Müller. Integral Probability Metrics and Their Generating Classes of Functions. *Advances in Applied Probability*, 29(2):429–443, 1997. 2, 4, 5

[41] 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. 1, 3, 6, 7, 8

[42] Thanh Thi Nguyen, Quoc Viet Hung Nguyen, Dung Tien Nguyen, Duc Thanh Nguyen, Thien Huynh-The, Saeid Nahavandi, Thanh Tam Nguyen, Quoc-Viet Pham, and Cuong M Nguyen. Deep learning for deepfakes creation and detection: A survey. *Computer Vision and Image Understanding*, 223:103525, 2022. 2

[43] Ehsan Nowroozi, Mauro Conti, and Yassine Mekdad. Detecting High-Quality GAN-Generated Face Images using Neural Networks, Mar. 2022. 3

[44] Augustus Odena, Vincent Dumoulin, and Chris Olah. Deconvolution and checkerboard artifacts. *Distill*, 2016. 1

[45] 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. 5

[46] Yuyang Qian, Guojun Yin, Lu Sheng, Zixuan Chen, and Jing Shao. Thinking in Frequency: Face Forgery Detection by Mining Frequency-aware Clues, Oct. 2020. arXiv:2007.09355 [cs]. 1

[47] Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. *CoRR*, abs/1511.06434, 2016. 5

[48] 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. 5

[49] Andreas Rossler, Davide Cozzolino, Luisa Verdoliva, Christian Riess, Justus Thies, and Matthias Nießner. Faceforensics++: Learning to detect manipulated facial images. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 1–11, 2019. 2

[50] 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. 2, 4

[51] Katja Schwarz, Yiyi Liao, and Andreas Geiger. On the Frequency Bias of Generative Models. In *Advances in Neural Information Processing Systems*, volume 34, pages 18126–18136. Curran Associates, Inc., 2021. 1

[52] Dongyao Shen, Youjian Zhao, and Chengbin Quan. Identity-Referenced Deepfake Detection with Contrastive Learning. In *Proceedings of the 2022 ACM Workshop on Information Hiding and Multimedia Security*, IH&MMSec '22, pages 27–32, New York, NY, USA, June 2022. Association for Computing Machinery. 1

[53] 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. 5, 6

[54] 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. 2, 4, 5

[55] 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. 6

[56] 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. 5

[57] 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. 5

[58] Laurens van der Maaten and Geoffrey E. Hinton. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9:2579–2605, 2008. 8

[59] 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. 1, 3, 6, 7, 8

[60] 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. 5

[61] Zhisheng Xiao, Karsten Kreis, and Arash Vahdat. Tackling the generative learning trilemma with denoising diffusion gans. 2022. 5, 6

[62] Ning Yu, Larry Davis, and Mario Fritz. Attributing fake images to gans: Learning and analyzing gan fingerprints. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 7555–7565, 2019. 1, 2, 3, 6, 7, 8

[63] 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. 6

[64] 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. 5

[65] Peng Zhou, Xintong Han, Vlad I. Morariu, and Larry S. Davis. Two-Stream Neural Networks for Tampered Face Detection, Mar. 2018. arXiv:1803.11276 [cs]. 1

[66] Xiangyu Zhu, Hao Wang, Hongyan Fei, Zhen Lei, and Stan Z. Li. Face Forgery Detection by 3D Decomposition. pages 2929–2939, 2021. 1