Towards the Detection of Diffusion Model Deepfakes

Jonas Ricker[a;*], **Simon Damm**[a], **Thorsten Holz**[b] **and Asja Fischer**[a]

[a]Ruhr University Bochum, Bochum, Germany
[b]CISPA Helmholtz Center for Information Security, Saarbrücken, Germany

Abstract. Diffusion models (DMs) have recently emerged as a promising method in image synthesis. However, to date, only little attention has been paid to the *detection* of DM-generated images, which is critical to prevent adverse impacts on our society. In this work, we address this pressing challenge from two different angles: First, we evaluate the performance of state-of-the-art detectors, which are very effective against images generated by generative adversarial networks (GANs), on a variety of DMs. Second, we analyze DM-generated images in the frequency domain and study different factors that influence the spectral properties of these images. Most importantly, we demonstrate that GANs and DMs produce images with different characteristics, which requires adaptation of existing classifiers to ensure reliable detection. We are convinced that this work provides the foundation and starting point for further research on effective detection of DM-generated images.

1 Introduction

In the recent past, diffusion models (DMs) have shown a lot of promise as a method for synthesizing images. Such models provide better (or at least similar) performance compared to generative adversarial networks (GANs) and enable powerful text-to-image models such as DALL-E 2 [33], Imagen [37], and Stable Diffusion [35]. Advances in image synthesis have resulted in very high quality images being generated, and humans can hardly tell if a given picture is an actual or artificially generated image (so-called *deepfake*) [32]. This progress has many implications in practice and poses a danger to our digital society: Deepfakes can be used for disinformation campaigns, as such images appear particularly credible due to their sensory comprehensibility. Disinformation aims to discredit opponents in public perception, to create sentiment for or against certain social groups, and thus influence public opinion. In their effect, deepfakes lead to an erosion of trust in institutions or individuals, support conspiracy theories, and promote a fundamental political camp formation. Despite the importance of this topic, there is only a limited amount of research on effective deepfake detection. Previous work on the detection of GAN-generated images (e.g., [51, 14, 28]) showed promising results, but it remains unclear if any of these methods are effective against DM-generated images.

In this paper, we present the first look at detection methods for DM-generated media. We tackle the problem from two different angles. On the one hand, we investigate whether DM-generated images can be effectively detected by existing methods that claim to be universal. We study ten models in total, five GANs and five DMs. We find that existing detection methods suffer from severe performance degradation when applied to DM-generated images, with the area under the receiver operating characteristic curve (AUROC) metric dropping by 15.2% on average compared to GANs. These results hint at a structural difference between synthetic images generated by GANs and DMs. We show that existing detection methods can be drastically improved by re-training them on DM-generated images. Interestingly, detectors trained on DM-generated images perform better on GAN-generated images than vice versa. Our results suggest that recognizing DM-generated images is a more difficult task than recognizing GAN images.

On the other hand, we analyze DM-generated images in the frequency domain and compare them to GAN-generated images. Unlike GANs, DMs do not exhibit grid-like artifacts in the frequency spectrum. However, we find a systematical mismatch towards higher frequencies. Further analysis suggests that the steps towards the end of the denoising process are most relevant for the high-frequency content, but at the same time the most challenging ones, which in turn may explain the frequency mismatch. We believe that our results provide the foundation for further research on the effective detection of deepfakes generated by DMs.

2 Related Work

Universal Fake Image Detection While in recent years a variety of successful methods to detect artificially generated images has been proposed [48], generalization to unseen data remains a challenging task [6]. Constructing an effective classifier for a specific generator is considered straightforward, which is why more research effort is put into designing universal detectors [56, 1, 51, 5, 14, 13, 28, 19]. This is especially important in the context of deceptive media, since new generative models emerge on a frequent basis and manually updating detectors is too slow to stop the propagation of harmful contents.

Frequency-Based Deepfake Detection Zhang et al. [58] were the first to demonstrate that the spectrum of GAN-generated images contains visible artifacts in the form of a periodic, grid-like pattern due to transposed convolution operations. These findings were later reproduced by Wang et al. [51] and extended to the discrete cosine transform (DCT) by Frank et al. [10]. Another characteristic was discovered by Durall et al. [8], who showed that GANs are unable to correctly reproduce the spectral distribution of the training data. In particular, generated images contain increased magnitudes at high frequencies. While several works attribute these spectral discrepancies to transposed convolutions [58, 8] or, more general, up-sampling operations [10, 2], no consensus on their origin has yet been reached. Some works explain them by the spectral bias of convolution layers

* Corresponding Author. Email: jonas.ricker@rub.de.

due to linear dependencies [9, 23], while others suggest the discriminator is not able to provide an accurate training signal [3, 40].

Detection of DM-Generated Images Despite the massive attention from the scientific community and beyond, DMs have not yet been sufficiently studied from the perspective of image forensics. A very specific use case is considered by Mandelli et al. [29], where the authors evaluate methods for detecting western blot images synthesized by different models, including DDPM [17]. More related to our analysis is the work of Wolter et al. [54], who propose to detect generated images based on their wavelet-packet representation, combining features from pixel- and frequency space. While the focus lies on GAN-generated images, they demonstrate that the images generated by ADM [7] can be detected using their approach. They also report that the classifier "appears to focus on the highest frequency packet", which is consistent with our findings. Jeong et al. [19] design a two-stage framework to train a universal detector from real images only. They train a fingerprint generator which reproduces frequency artifacts of generative models, which is then used to create the training data for the detector. Their method achieves high detection accuracy for GAN-generated images and more recent models, including DDPM [17].

Frequency Considerations in DMs Both Kingma et al. [24] and Song et al. [45] experiment with adding Fourier features to improve learning of high-frequency content, the former reporting it leads to much better likelihoods. Another interesting observation is made by Rissanen et al. [34] who analyze the generative process of diffusion models in the frequency domain. They state that diffusion models have an inductive bias according to which, during the reverse process, higher frequencies are added to existing lower frequencies.

3 Background on DMs

DMs were first proposed by Sohl-Dickstein et al. [41] and later advanced by Ho et al. [17]. Song et al. later pointed out the connections between DMs and score-based generative models [43, 44, 45]. Since then, numerous modifications and improvements have been proposed, leading to higher perceptual quality [31, 7, 4, 35] and increased sampling speed [42, 25, 38, 55]. In short, a DM models a data distribution by gradually disturbing a sample from this distribution and then learning to reverse this diffusion process. In the diffusion (or forward) process, a sample $\mathbf{x}_0$ (an image in most applications) is repeatedly corrupted by Gaussian noise in sequential steps $t = 1, \dots, T$:

$$q(\mathbf{x}_t|\mathbf{x}_{t-1}) = \mathcal{N}(\mathbf{x}_t; \sqrt{1-\beta_t}\mathbf{x}_{t-1}, \beta_t\mathbf{I}) \ , \tag{1}$$

where $\{\beta_t\}_{t=1}^T$ defines the noise schedule. With $\alpha_t \coloneqq 1 - \beta_t$ and $\bar{\alpha}_t \coloneqq \prod_{s=1}^{t} \alpha_t$, we can directly sample from the forward process at arbitrary time steps t via

$$q(\mathbf{x}_t|\mathbf{x}_0) = \mathcal{N}(\mathbf{x}_t; \sqrt{\bar{\alpha}_t}\mathbf{x}_0, (1-\bar{\alpha}_t)\mathbf{I}) \ . \tag{2}$$

During the denoising (or reverse) process, we aim to sample from $q(\mathbf{x}_{t-1}|\mathbf{x}_t)$ to ultimately obtain a clean image given $\mathbf{x}_T$. However, since $q(\mathbf{x}_{t-1}|\mathbf{x}_t)$ is intractable as it depends on the entire underlying data distribution, it is approximated by a deep neural network. More formally, $q(\mathbf{x}_{t-1}|\mathbf{x}_t)$ is approximated by

$$p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t) = \mathcal{N}(\mathbf{x}_{t-1}; \mu_\theta(\mathbf{x}_t, t), \Sigma_\theta(\mathbf{x}_t, t)) \ , \tag{3}$$

where mean μ_θ and covariance Σ_θ are given by the output of the model (or the latter is set to a constant as proposed by Ho et al. [17]). Predicting the mean of the denoised sample $\mu_\theta(\mathbf{x}_t, t)$ is conceptually equivalent to predicting the noise that should be removed, denoted by $\epsilon_\theta(\mathbf{x}_t, t)$. Predominantly, the latter approach is implemented (e.g., [17, 7]) such that training a DM boils down to minimizing a (weighted) mean-squared error (MSE) $\|\epsilon - \epsilon_\theta(\mathbf{x}_t, t)\|^2$ between the true and predicted noise.

4 Experimental Setup

An overview of the models under evaluation is shown in Table 1, and example images are given in Supplement A. We consider data from ten models in total, five GANs and five DMs. This includes the seminal models ProGAN and StyleGAN, as well as the more recent ProjectedGAN. Note that the recently proposed methods, Diff(usion)-StyleGAN2 and Diff(usion)-ProjectedGAN (the current state of the art on LSUN Bedroom) use a forward diffusion process to optimize GAN training, but this does not change the GAN model architecture. From the class of DMs, we consider the original DDPM, its successor IDDPM, and ADM, the latter outperforming several GANs with an Fréchet inception distance (FID) of 1.90 on LSUN Bedroom. PNDM optimizes the sampling process by the factor of 20 using pseudo numerical methods, which can be applied to existing pre-trained DMs. Lastly, LDM uses an adversarially trained autoencoder that transforms an image from the pixel space to a latent space (and back). Training the DM in this more suitable latent space reduces the computational complexity and therefore enables training on higher resolutions. The success of this approach is underpinned by the recent publication of Stable Diffusion, a powerful and publicly available text-to-image model based on LDM.

Table 1. Models evaluated in this work. FIDs are taken from the original publications and from [7] in the case of IDDPM. A lower FID corresponds to higher image quality.

Model Class	Method	FID on LSUN Bedroom
GAN	ProGAN [20]	8.34
	StyleGAN [21]	2.65
	ProjectedGAN [39]	1.52
	Diff-StyleGAN2 [52]	3.65
	Diff-ProjectedGAN [52]	1.43
DM	DDPM [17]	6.36
	IDDPM [31]	4.24
	ADM [7]	1.90
	PNDM [25]	5.68
	LDM [35]	2.95

All models are trained on LSUN Bedroom with a resolution of 256×256. Samples are either directly downloaded or generated using code and pre-trained models provided by the original publications. More detailed descriptions are given in Supplement A. For each model, we collect 50k samples in total from which 39k are used for training, 1k for validation, and 10k for testing and frequency analysis, if not stated otherwise.

In this work, we mainly perform experiments with generative models trained on LSUN Bedroom, since it is the only dataset for which model checkpoints and/or samples from *all* GANs and DMs are available. This allows for a fair comparison between generators by avoiding any biases due to the complexity and spectral properties of different datasets. However, we additionally evaluate the performance of state-of-the-art detectors on a variety of datasets including FFHQ [21], ImageNet [36], multiple LSUN [57] classes, and images

Table 2. Detection performance of pre-trained universal detectors. For the detectors of Wang et al. [51] and Gragnaniello et al. [14], we consider two different variants, respectively. In the upper half, we report the performance on models trained on LSUN Bedroom, while results on additional datasets are given in the second half. The best score (determined by the highest Pd@1%) for each generator is highlighted in **bold**. We report average scores in gray.

AUROC / Pd@5% / Pd@1%	Wang et al. [51]						Gragnaniello et al. [14]						Mandelli et al. [28]		
	Blur+JPEG (0.5)			Blur+JPEG (0.1)			ProGAN			StyleGAN2					
ProGAN	**100.0** / **100.0** / **100.0**			**100.0** / **100.0** / **100.0**			**100.0** / **100.0** / **100.0**			**100.0** / **100.0** / **100.0**			91.2 /	54.6 /	27.5
StyleGAN	98.7 /	93.7 /	81.4	99.0 /	95.5 /	84.4	**100.0** / **100.0** / **100.0**			**100.0** / **100.0** / **100.0**			89.6 /	43.6 /	14.7
ProjectedGAN	94.8 /	73.8 /	49.1	90.9 /	61.8 /	34.5	**100.0** / **99.9** / **99.3**			99.9 /	99.6 /	97.8	59.4 /	8.4 /	2.4
Diff-StyleGAN2	99.9 /	99.6 /	97.9	100.0 /	99.9 /	99.3	**100.0** / **100.0** / **100.0**			**100.0** / **100.0** / **100.0**			100.0 /	100.0 /	99.9
Diff-ProjectedGAN	93.8 /	69.5 /	43.3	88.8 /	54.6 /	27.2	**99.9** / **99.9** / **99.2**			99.8 /	99.6 /	96.6	62.1 /	10.5 /	2.8
Average	97.4 /	87.3 /	74.3	95.7 /	82.4 /	69.1	**100.0** / **100.0** / **99.7**			99.9 /	99.8 /	98.9	80.4 /	43.4 /	29.5
DDPM	85.2 /	37.8 /	14.2	80.8 /	29.6 /	9.3	**96.5** / **79.4** / **39.1**			95.1 /	69.5 /	30.7	57.4 /	3.8 /	0.6
IDDPM	81.6 /	30.6 /	10.6	79.9 /	27.6 /	7.8	**94.3** / **64.8** / **25.7**			92.8 /	58.0 /	21.2	62.9 /	7.0 /	1.3
ADM	68.3 /	13.2 /	3.4	68.8 /	14.1 /	4.0	**77.8** / **20.7** / **5.2**			70.6 /	13.0 /	2.5	60.5 /	8.2 /	1.8
PNDM	79.0 /	27.5 /	9.2	75.5 /	22.6 /	6.3	91.6 /	52.0 /	16.6	**91.5** / **53.9** / **22.2**			71.6 /	15.4 /	4.0
LDM	78.7 /	24.7 /	7.4	77.7 /	24.3 /	6.9	96.7 /	79.9 /	42.1	**97.0** / **81.8** / **48.9**			54.8 /	7.7 /	2.1
Average	78.6 /	26.8 /	9.0	76.6 /	23.7 /	6.8	**91.4** / **59.3** / **25.7**			89.4 /	55.2 /	25.1	61.4 /	8.4 /	2.0
ADM (LSUN Cat)	58.4 /	8.4 /	2.5	58.1 /	8.5 /	3.3	**60.2** / **9.3** / **4.2**			51.7 /	5.5 /	1.8	55.6 /	6.0 /	1.3
ADM (LSUN Horse)	55.5 /	6.7 /	1.5	53.4 /	6.0 /	2.2	**56.1** / **7.5** / **2.7**			50.2 /	4.8 /	1.4	44.2 /	2.4 /	0.5
ADM (ImageNet)	69.1 /	13.9 /	4.1	71.7 /	15.5 /	4.5	72.1 /	13.2 /	3.5	**83.9** / **38.5** / **16.6**			60.1 /	6.8 /	0.9
ADM-G-U (ImageNet)	67.2 /	11.7 /	3.7	62.3 /	6.2 /	1.2	66.8 /	6.0 /	1.6	**78.9** / **26.7** / **10.2**			60.0 /	7.6 /	1.0
PNDM (LSUN Church)	76.9 /	25.7 /	10.2	77.6 /	28.4 /	12.0	90.9 /	54.1 /	24.5	**99.3** / **96.5** / **85.8**			56.4 /	6.9 /	1.9
LDM (LSUN Church)	86.3 /	42.1 /	19.8	82.2 /	33.9 /	14.2	98.8 /	93.7 /	75.5	**99.5** / **98.2** / **90.2**			58.9 /	5.8 /	1.3
LDM (FFHQ)	69.4 /	14.2 /	3.6	71.0 /	14.8 /	3.6	**91.1** / **54.2** / **25.4**			67.2 /	10.2 /	2.1	63.0 /	5.6 /	0.6
ADM' (FFHQ)	77.7 /	24.7 /	8.7	81.4 /	28.2 /	8.8	**87.7** / **41.9** / **17.8**			89.0 /	45.5 /	17.2	69.8 /	11.1 /	2.0
P2 (FFHQ)	79.5 /	26.4 /	8.9	83.2 /	30.1 /	9.2	89.2 /	40.5 /	11.5	**91.1** / **51.9** / **18.9**			72.5 /	13.6 /	2.7
Stable Diffusion v1-1	42.4 /	4.0 /	1.5	51.4 /	6.9 /	2.0	73.2 /	13.3 /	4.0	**75.2** / **28.1** / **13.6**			76.1 /	21.2 /	4.2
Stable Diffusion v1-5	43.7 /	4.1 /	1.4	52.6 /	7.0 /	2.1	72.9 /	11.1 /	2.8	**79.8** / **35.6** / **18.3**			75.3 /	21.1 /	4.1
Stable Diffusion v2-1	**46.1** /	**4.7** /	**1.4**	47.3 /	5.5 /	1.1	62.8 /	6.9 /	1.1	55.1 /	5.1 /	1.1	37.0 /	3.2 /	0.5
Midjourney v5	52.7 /	9.3 /	3.0	57.1 /	11.3 /	3.0	**69.9** / **12.7** / **3.3**			67.1 /	12.3 /	3.3	18.3 /	1.0 /	0.3

generated by the popular text-to-image models Stable Diffusion and Midjourney (details in Supplement D.1). Moreover, we provide the frequency analysis on these additional datasets in Supplement D.2.

We release our code and dataset at https://github.com/jonasricker/diffusion-model-deepfake-detection.

5 Detection of DM-Generated Images

We now investigate the effectiveness of state-of-the-art fake image detectors on DM-generated images. First, we analyze the performance of off-the-shelf pre-trained models which are highly capable of detecting images generated by GANs, and then perform experiments with models re-trained or fine-tuned on DM-generated images. We also show example images considered more or less "fake" based on the detector's output in Supplement B.5.

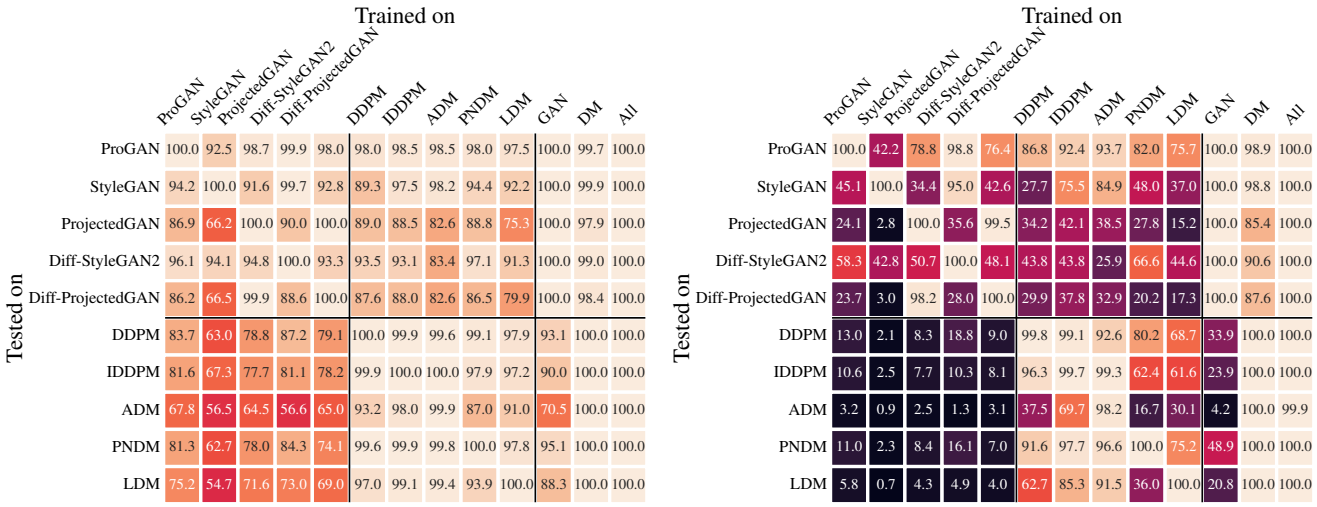
In this work, we evaluate three state-of-the-art detection methods by Wang et al. [51], Gragnaniello et al. [14], and Mandelli et al. [28]. A description of these detectors is provided in Supplement B.1. All three claim to perform well on unseen data, but it is unclear whether this holds for DM-generated images as well. The performance of the analyzed classifiers is estimated in terms of the widely used AUROC. As pointed out by Cozzolino et al. [5], however, the AUROC is overly optimistic as it captures merely the potential of a classifier, but the optimal threshold is usually unknown. Thus, we adopt the use of the probability of detection at a fixed false alarm rate (Pd@FAR) as an additional metric, which is given as the true positive rate at a fixed false positive rate. Intuitively, this corresponds to picking the y-coordinate of the ROC curve given an x-coordinate. This metric is a valid choice for realistic scenarios such as large-scale content filtering on social media, where only a small amount of false positives is tolerable. We consider FARs of 5% and 1%.

5.1 *Can we use State-of-the-Art Detectors for DM-Generated Images?*

Our first analysis aims to answer whether DM-generated images are similar, with respect to detectable traces, to those generated by GANs. If this was the case, we would expect universal fake image detectors trained on GANs to achieve a high detection accuracy.

Using the dataset introduced in Section 4, we analyze the performance of the three state-of-the-art detectors. We use pre-trained models provided by the authors without any modifications and compute the evaluation metrics using 10k real and 10k generated images. The results are given in the upper half of Table 2. For GAN-generated images, both classifiers provided by Gragnaniello et al. [14] achieve almost perfect detection results across all five generators, even w.r.t. the challenging Pd@1% metric. For images generated by ProjectedGAN and Diff-ProjectedGAN, the variant trained on ProGAN images performs slightly better. In comparison, the performance of the classifier provided by Wang et al. [51] is reduced (except for ProGAN, on which the model was trained). Similarly, ProjectedGAN and Diff-ProjectedGAN achieve the lowest scores, with Pd@1% dropping below 50%. The detection method proposed by Mandelli et al. [28] achieves significantly worse results than its contestants, except for Diff-StyleGAN2. It should be noted that StyleGAN2 images make up a large portion of the training data for this classifier.

For images generated by DMs, we observe a sharp deterioration for all detectors, with AUROC dropping by 15.2% on average in comparison to GANs. The difference becomes even more severe when looking at the more realistic Pd@FAR metrics. While the best model by Gragnaniello et al. [14] (trained on ProGAN images) achieves an average Pd@5% of 100.0% and Pd@1% of 99.7% on GAN-generated images, the scores drop to 59.3% and 25.7% for DMs, respectively. Given a setting where a low FAR is required, we find that none of the detectors under evaluation is usable. Among the



(a) AUROC (b) Pd@1%

Figure 1. Detection performance for detectors trained from scratch. The columns *GAN*, *DM*, and *All* correspond to models trained on samples from all GANs, DMs, and both, respectively.

five DMs, ADM-generated images are the most difficult to detect, possibly due to the low FID (see Table 1). However, images from LDM, which achieves the second lowest FID (among DMs), can be detected quite effectively. A potential reason for this could be the adversarial training procedure of LDM, which may result in properties similar to those of GAN-generated images.

To verify that our findings do not depend on the selected dataset, we apply the detectors to DMs trained on a variety of other datasets. Additionally, we consider images generated by the popular text-to-image models Stable Diffusion and Midjourney, which recently attracted both scientific and public interest. Details on these datasets are provided in Supplement D. The results, given in the lower half of Table 2, support the finding that detectors perform significantly worse on DM-generated images. Both PNDM and LDM trained on LSUN Church, however, are detected significantly better by all classifiers except the detector from Mandelli et al. [28]. This might be caused by the relatively high FID both models achieve on this dataset (8.69 for PNDM, 4.02 for LDM) compared to the state of the art (1.59 by ProjectedGAN [39]).

Overall, we observe that pre-trained universal detectors suffer from a severe performance drop when applied to DM-generated images (compared to GANs). In an extended experiment (see Supplement B.4) we also find that image perturbations, like compression or blurring, have a stronger adverse effect on the detectability of DM-generated images. We conclude that there exists a structural difference between GAN- and DM-generated images that requires the adaptation of existing methods or the development of novel methods to ensure the reliable detection of synthesized images.

5.2 *Training Detectors on DM-Generated Images*

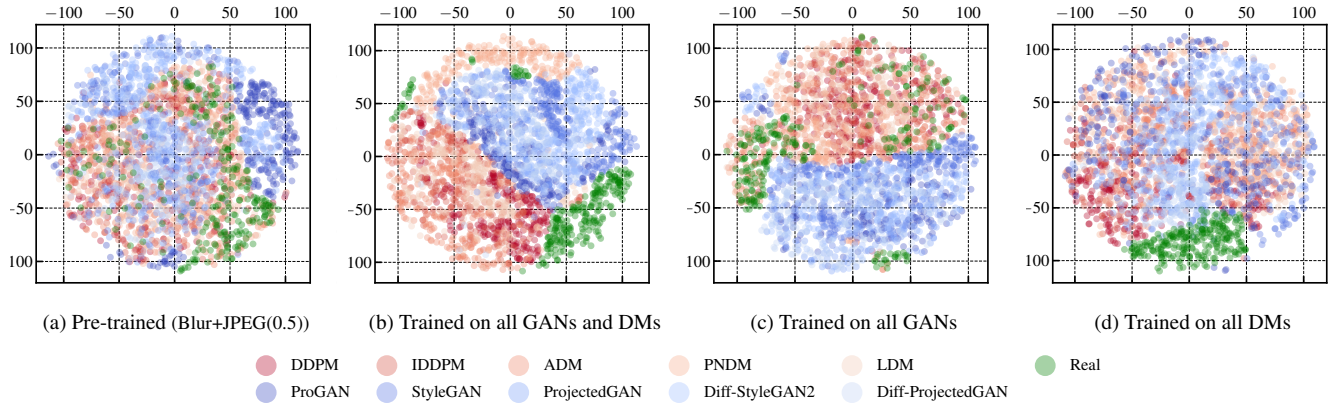
Given the findings presented above, the question remains whether DMs evade detection in principle, or whether the detection performance can be increased by re-training a detector from scratch.[1] We select the architecture and training procedure from Wang et al. [51] since the original training code is available[2]. Furthermore, we choose the configuration Blur+JPEG (0.5) as it yields slightly better scores on average, and use the validation set for adapting the learning rate and performing early stopping. Besides training a model on data from each generator, we also consider three aggregated settings in which we train on all images generated by GANs, DMs, and both, respectively.

The results are provided in Figure 1. A model which is trained on images from a specific DM achieves near-perfect scores, with Pd@1% ranging from 98.2% for ADM to 100.0% for PNDM and LDM. To a limited extent, detectors are capable of generalizing to other DMs, indicating common, detectable characteristics. Interestingly, detectors trained on images from DMs are significantly more successful in detecting GAN-generated images than vice versa. The same holds for models trained using images from multiple GANs and DMs, respectively.

To deepen our understanding we analyze how training on different data affects the feature spaces of the detectors. We utilize t-SNE [47] to visualize the extracted features prior to the last fully-connected layer in Figure 2. For the pre-trained detector by Wang et al. [51] we observe a relatively clear separation between real and GAN-generated images, while there exists a greater overlap between real and DM-generated images (Figure 2a). These results match the classification results from Table 2. Looking at the detector which is trained on DM-generated images only (Figure 2d), the feature representations for GAN- and DM-generated images appear to be similar. In contrast, the detectors trained using GAN-generated images or both (Figures 2c and 2b) seem to learn distinct feature representations for GAN- and DM-generated images. Based on these results we hypothesize that GANs and DMs generate images with some common, detectable characteristics, which a detector trained on DM-generated images only is able to capture. However, detectors trained on GAN-generated images appear to focus mostly on GAN-specific patterns, which may be more prominent and therefore easier to detect. Such patterns could be frequency artifacts, as shown in Section 6. An extended feature space analysis including MMD [15] is

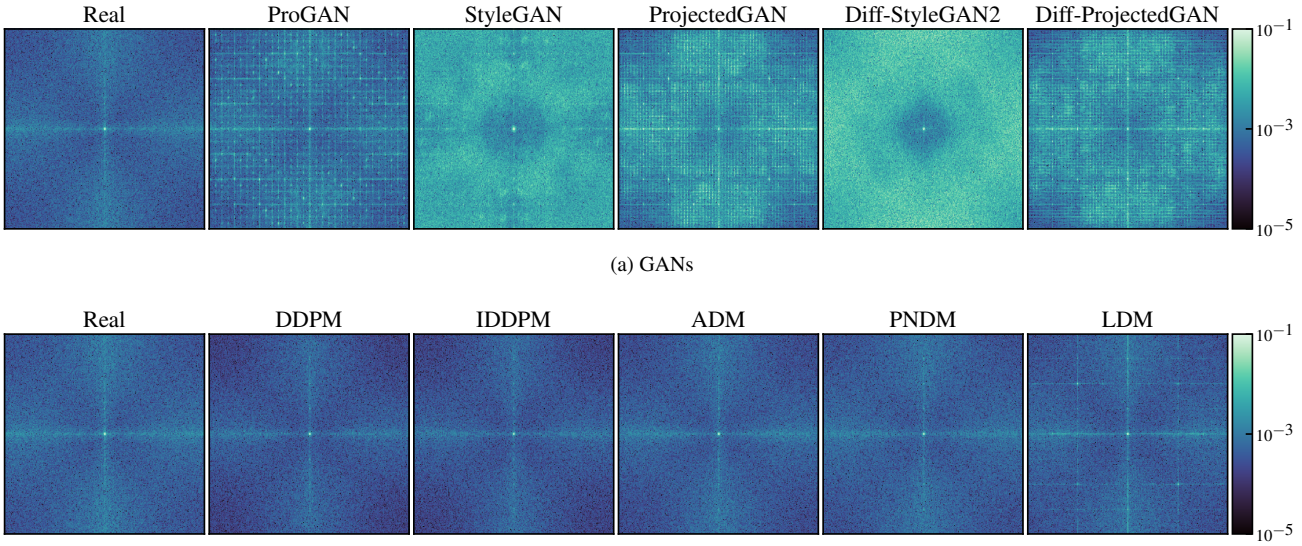
[1] Note that here "from scratch" means that we start from a ResNet-50 model pre-trained on ImageNet, which is exactly how Wang et al. [51] trained their models. We carry out the same evaluation for fine-tuned detectors in Supplement B.2, which leads to very similar results.

[2] https://github.com/peterwang512/CNNDetection



(a) Pre-trained (Blur+JPEG(0.5)) (b) Trained on all GANs and DMs (c) Trained on all GANs (d) Trained on all DMs

Figure 2. Feature space visualization for the detector by Wang et al. [51] via t-SNE of real and generated images (LSUN Bedroom) in two dimensions. The features $f(\cdot)$ correspond to the representation prior to the last fully-connected layer of the respective detection method, which are of dimension 2048. Additional results are given in Figure 11 in Supplement B.3.



(a) GANs

(b) DMs

Figure 3. Mean DFT spectrum of real and generated images. To increase visibility, the color bar is limited to $[10^{-5}, 10^{-1}]$, with values lying outside this interval being clipped.

provided in Supplement B.3.

Notably, the results demonstrate that training a detector which is capable of detecting both GAN- and DM-generated images is clearly possible (see rightmost column of Figure 1a). While this finding is promising, it remains unclear whether a similar performance can be achieved in the model- and dataset-agnostic setting addressed by Wang et al. [51].

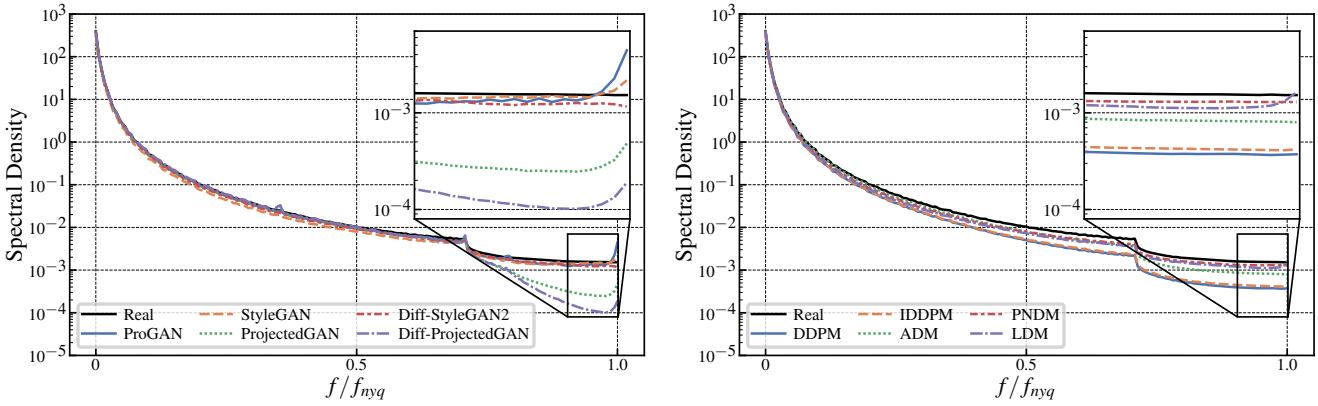
6 Analysis of DM-Generated Images in the Frequency Domain

The visual quality of synthesized images has reached a level that makes them practically indistinguishable from real images for humans [32]. However, in the case of GANs, the same does not hold when switching to the frequency domain [58]. In this section, we analyze the spectral properties of DM-generated images and compare them to those of GAN-generated images. We use three frequency transforms that have been applied successfully in both traditional image forensics [27] and deepfake detection: discrete Fourier transform (DFT), discrete cosine transform (DCT), and the reduced spectrum, which can be viewed as a 1D representation of the DFT. While DFT and DCT visualize frequency artifacts, the reduced spectrum can be used to identify spectrum discrepancies. The formal definitions of all transforms are provided in Supplement C.1. Furthermore, we evaluate the influence of different parameters during training and sampling of DMs and hypothesize on the origin of spectrum discrepancies.

6.1 *Spectral Properties of DM- vs GAN-Generated Images*

Figure 3 depicts the average absolute DFT spectrum of 10k images from each GAN and DM, each trained on LSUN Bedroom[3]. Before

[3] Due to space limitations we provide the DCT spectra in Supplement C.3 and the analysis on additional datasets in Supplement D.2.



(a) GANs (b) DMs

Figure 4. Mean reduced spectrum of real and generated images. The part of the spectrum where GAN-characteristic discrepancies occur is magnified.

applying the DFT, images are transformed to grayscale and, following previous works [30, 51], high-pass filtered by subtracting a median-filtered version of the image. For all GANs we observe significant artifacts, predominantly in the form of a regular grid. Taking the reduced spectra in Figure 4a into account we confirm the previously reported elevated high frequencies (see Section 2). For ProjectedGAN and Diff-ProjectedGAN, this characteristic is also visible. However, the spectral density does not exceed that of real images. Diff-StyleGAN2 is an exception as it shows a slight decrease towards the end of the spectrum.

In contrast, the DFT spectra of images generated by DMs (see Figure 3b), except LDM, are significantly more similar to the real spectrum with almost no visible artifacts. LDM, however, exhibits a thin grid across its spectrum, as well as the increased amount of high frequencies which is characteristic for GANs (see Figure 4b). As mentioned in Section 4, the architecture of LDM differs from the remaining DMs as the actual image is generated using an adversarially trained autoencoder, which could explain the discrepancies. This supports previous findings which suggest that the discriminator is responsible for spectrum deviations [3, 40]. In Figure 5, we analyze whether this observation also holds for images generated by Stable Diffusion, which is based on LDM. Upon close inspection one can see a set of bright dots arranged in a rectangular grid. Interestingly, these artifacts are less pronounced in images generated by Stable Diffusion 2-1, which could hint towards higher image quality.

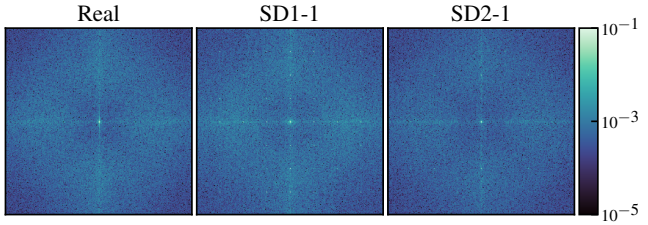


Figure 5. Mean DFT spectrum of real images and images generated by Stable Diffusion 1-1 and 2-1. To increase visibility, the color bar is limited to $[10^{-5}, 10^{-1}]$, with values lying outside this interval being clipped.

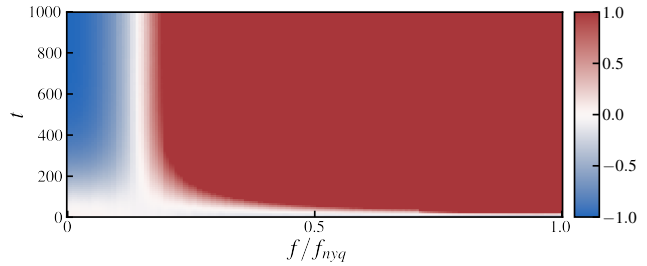
Although the reduced spectra of most DMs (see Figure 4b) do not exhibit the GAN-like rise towards the end of the spectrum, several DMs deviate stronger from the spectrum of real images than GANs. Especially for DDPM, IDDPM, and ADM we observe an underestimation of the spectral density which becomes stronger towards higher frequencies (relative to the overall spectral density level). We discuss possible causes for this in the following section.

In a complementary experiment (see Supplement C.2) inspired by Frank et al. [10] we try to answer whether the reduced amount of frequency artifacts makes DM-generated images less detectable in the frequency domain. On the one hand, using a simple logistic regression classifier, DM-generated images can be better classified in the frequency space than in pixel space. On the other hand, the detection accuracy in both frequency and pixel space is significantly lower compared to GAN-generated images, which makes it difficult to draw conclusions. Nevertheless, the fact that a simple classifier performs significantly better on GAN-generated images strengthens the hypothesis that they exhibit more pronounced generation traces than DM-generated images.

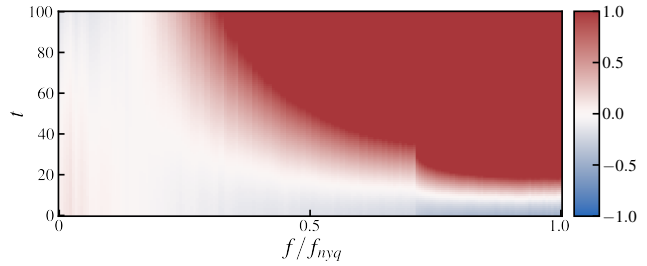
6.2 *Investigating the Source of Spectrum Discrepancies*

As explained in Section 2, several recent works attempt to identify the source of spectrum discrepancies in GAN-generated images. In this section, we make an effort to launch a similar line of work for DMs. For the experiments, we used code and model from ADM [7] trained on LSUN Bedroom. All spectra are generated using 512 grayscale images.

Since DMs generate images via gradual denoising, we analyze how the spectrum evolves during this denoising process. We generate samples at different time steps t and compare their average reduced spectrum to that of 50k real images. The results are shown in Figure 6. We adopt the figure type from Schwarz et al. [40] and depict the relative spectral density error $\tilde{S}_{\text{err}} = \tilde{S}_{\text{fake}}/\tilde{S}_{\text{real}} - 1$, with the colorbar clipped at -1 and 1. At $t = 1000$, the image is pure Gaussian noise, which causes the strong spectrum deviations. Around $t = 300$, the error starts to decrease, but interestingly it appears to not continuously improve until $t = 0$. With regard to high frequencies, we observe an optimum at approximately $t = 10$ followed by an increasing underestimation. In Supplement C.4, we additionally provide the spectral density error between the denoising process and the diffusion process at the corresponding step, which further visualizes this behavior. However, at $t = 10$, the images are easier to detect and also visually noisier (see Supplement C.4 Figures 17 and 18). Thus, stopping the denoising process early is not an effective means to yield less detectable images.



(a) $0 \leq t \leq 1000$



(b) $0 \leq t \leq 100$

Figure 6. Evolution of spectral density error throughout the denoising process. The error is computed relatively to the spectrum of real images. We display the error for (a) all sampling steps and (b) a close-up of the last 100 steps. The colorbar is clipped at -1 and 1.

We hypothesize that the underestimation towards higher frequencies is due to the denoising capability of the model at low noise levels. The full line of thoughts builds on observations from research on denoising autoencoders and is given in Supplement C.6, here we briefly summarize it. In this context the training objective determines the importance of denoising tasks at different noise levels. The objective proposed by Ho et al. [17], $L_{\text{simple}} = \mathbb{E}_{t,\mathbf{x}_0,\epsilon}[\|\epsilon - \epsilon_\theta(\mathbf{x}_t, t)\|^2]$, for example, considers each denoising task as equally important. As noted by Nichol et al. [31], this objective does not lead to good likelihood values. The latter depend on high-frequency details synthesized at the denoising tasks near $t = 0$ [24].

However, removing these noise artifacts is clearly more challenging as evident from the MSE in Figure 7. The classical variational lower bound L_{vlb} would force the model to focus on these challenging denoising steps towards $t = 0$. Since L_{vlb} is extremely difficult or even impossible to optimize (see e.g., [31]) the hybrid objective $L_{\text{hybrid}} = L_{\text{simple}} + \lambda L_{\text{vlb}}$ (with $\lambda = 0.001$) is proposed by Nichol et al. [31] and also employed by Dhariwal et al. [7]. We conclude that the denoising steps at low noise levels, which govern the high-frequency content of generated images, are the most difficult denoising steps. By down-weighting the loss at these steps (relatively to the L_{vlb}), DMs achieve remarkable perceptual image quality, but seem to fall short on matching the high-frequency distribution of real data. We elaborate in Supplement C.6.

In an additional experiment we also investigate how the number of sampling steps influences the spectral properties of generated images (see Supplement C.5). As should be expected, more sampling steps lead to higher image quality and lower frequency errors at the cost of longer computation.

7 Conclusion

In this work, we make a much-needed first step towards the detection of DM-generated images and open up a new chapter in the field

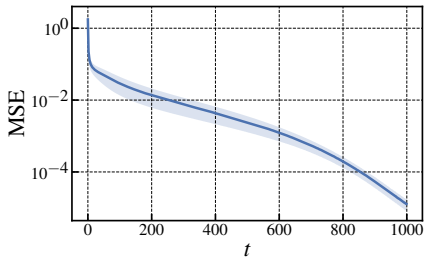


Figure 7. Mean and standard deviation of the MSE $\|\epsilon - \epsilon_\theta(\mathbf{x}_t, t)\|^2$. The denoising tasks towards $t = 0$, accounting for high frequencies, are more difficult.

of synthetic image forensics. We find that existing detectors, which claim to be universal, fail to effectively recognize images synthesized by different DMs. This hints at fundamental differences between images generated by DMs and GANs. However, existing detectors can be adapted to DMs by re-training, which demonstrates that DM-generated images *can* in fact be detected. Remarkably, detectors trained on DM-generated images generalize better to GAN-generated images than vice versa. Based on the results of our feature space analysis, we speculate that images generated by DMs contain patterns which are similar to those of GAN-generated images, but less pronounced. Developing novel detection techniques for DM-generated images could therefore benefit the detection of synthetic images altogether.

While DMs produce little to no grid-like frequency artifacts known from GANs, there appears to be a systematic mismatch towards higher frequencies. Our hypothesis is that during training, less weight is attached to these frequencies due to the choice of the training objective. However, this is an appropriate choice because correctly reproducing high frequencies is less important to the perceived quality of generated images than matching lower frequencies. Whether spectral discrepancies can be reduced (e.g., by adding spectral regularization [8]) without impairing image quality should be the subject of future work.

Most importantly, more efforts are needed to develop novel detection methods, both universal and DM-specific, to fight harmful visual disinformation. This is especially relevant due to the emergence of powerful text-to-image models based on DMs like Stable Diffusion [35]. We hope that our work can spark further research in this direction and emphasizes the fact that advances in realistic media synthesis should always be accompanied by considerations regarding deepfake detection methods.

Acknowledgements

Funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany´s Excellence Strategy - EXC 2092 CASA – 390781972.

References

[1] Lucy Chai, David Bau, Ser-Nam Lim, and Phillip Isola, 'What makes fake images detectable? Understanding properties that generalize', in *European Conference on Computer Vision (ECCV)*, (2020).

[2] Keshigeyan Chandrasegaran, Ngoc-Trung Tran, and Ngai-Man Cheung, 'A closer look at Fourier spectrum discrepancies for CNN-generated images detection', in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, (2021).

[3] Yuanqi Chen, Ge Li, Cece Jin, Shan Liu, and Thomas Li, 'SSD-GAN: Measuring the realness in the spatial and spectral domains', in *AAAI Conference on Artificial Intelligence (AAAI)*, (2021).

[4] Jooyoung Choi, Jungbeom Lee, Chaehun Shin, Sungwon Kim, Hyunwoo Kim, and Sungroh Yoon, 'Perception prioritized training of diffusion models', in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, (2022).

[5] Davide Cozzolino, Diego Gragnaniello, Giovanni Poggi, and Luisa Verdoliva, 'Towards universal GAN image detection', in *International Conference on Visual Communications and Image Processing (VCIP)*, (2021).

[6] Davide Cozzolino, Justus Thies, Andreas Rössler, Christian Riess, Matthias Nießner, and Luisa Verdoliva, 'ForensicTransfer: Weakly-supervised domain adaptation for forgery detection', *Computing Research Repository (CoRR)*, (2019).

[7] Prafulla Dhariwal and Alexander Nichol, 'Diffusion models beat GANs on image synthesis', in *Advances in Neural Information Processing Systems (NeurIPS)*, (2021).

[8] Ricard Durall, Margret Keuper, and Janis Keuper, 'Watch your up-convolution: CNN based generative deep neural networks are failing to reproduce spectral distributions', in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, (2020).

[9] Tarik Dzanic, Karan Shah, and Freddie Witherden, 'Fourier spectrum discrepancies in deep network generated images', in *Advances in Neural Information Processing Systems (NeurIPS)*, (2020).

[10] Joel Frank, Thorsten Eisenhofer, Lea Schönherr, Asja Fischer, Dorothea Kolossa, and Thorsten Holz, 'Leveraging frequency analysis for deep fake image recognition', in *International Conference on Machine Learning (ICML)*, (2020).

[11] Krzysztof J. Geras and Charles Sutton, 'Scheduled denoising autoencoders', in *International Conference on Learning Representations (ICLR)*, (2015).

[12] Krzysztof J. Geras and Charles Sutton, 'Composite denoising autoencoders', in *European Conference on Machine Learning and Knowledge Discovery in Databases (ECML PKDD)*, (2016).

[13] Sharath Girish, Saksham Suri, Sai Saketh Rambhatla, and Abhinav Shrivastava, 'Towards discovery and attribution of open-world GAN generated images', in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, (2021).

[14] D. Gragnaniello, D. Cozzolino, F. Marra, G. Poggi, and L. Verdoliva, 'Are GAN generated images easy to detect? A critical analysis of the state-of-the-art', in *IEEE International Conference on Multimedia and Expo (ICME)*, (2021).

[15] Arthur Gretton, Karsten M. Borgwardt, Malte J. Rasch, Bernhard Schölkopf, and Alexander Smola, 'A kernel two-sample test', *Journal of Machine Learning Research (JMLR)*, **13**(25), 723–773, (2012).

[16] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, 'Deep residual learning for image recognition', in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, (2016).

[17] Jonathan Ho, Ajay Jain, and Pieter Abbeel, 'Denoising diffusion probabilistic models', in *Advances in Neural Information Processing Systems (NeurIPS)*, (2020).

[18] Nils Hulzebosch, Sarah Ibrahimi, and Marcel Worring, 'Detecting CNN-generated facial images in real-world scenarios', in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, (2020).

[19] Yonghyun Jeong, Doyeon Kim, Youngmin Ro, Pyounggeon Kim, and Jongwon Choi, 'FingerprintNet: Synthesized fingerprints for generated image detection', in *European Conference on Computer Vision (ECCV)*, (2022).

[20] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen, 'Progressive growing of GANs for improved quality, stability, and variation', in *International Conference on Learning Representations (ICLR)*, (2018).

[21] Tero Karras, Samuli Laine, and Timo Aila, 'A style-based generator architecture for generative adversarial networks', in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, (2019).

[22] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila, 'Analyzing and improving the image quality of StyleGAN', in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, (2020).

[23] Mahyar Khayatkhoei and Ahmed Elgammal, 'Spatial frequency bias in convolutional generative adversarial networks', *AAAI Conference on Artificial Intelligence (AAAI)*, (2022).

[24] Diederik Kingma, Tim Salimans, Ben Poole, and Jonathan Ho, 'Variational diffusion models', in *Advances in Neural Information Processing Systems (NeurIPS)*, (2021).

[25] Luping Liu, Yi Ren, Zhijie Lin, and Zhou Zhao, 'Pseudo numerical methods for diffusion models on manifolds', in *International Conference on Learning Representations (ICLR)*, (2022).

[26] Zhengzhe Liu, Xiaojuan Qi, and Philip H. S. Torr, 'Global texture enhancement for fake face detection in the wild', in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, (2020).

[27] Siwei Lyu, *Natural Image Statistics in Digital Image Forensics*, Ph.D. dissertation, Dartmouth College, 2008.

[28] Sara Mandelli, Nicolò Bonettini, Paolo Bestagini, and Stefano Tubaro, 'Detecting GAN-generated images by orthogonal training of multiple CNNs', in *IEEE International Conference on Image Processing (ICIP)*, (2022).

[29] Sara Mandelli, Davide Cozzolino, Edoardo D. Cannas, João P. Cardenuto, Daniel Moreira, Paolo Bestagini, Walter J. Scheirer, Anderson Rocha, Luisa Verdoliva, Stefano Tubaro, and Edward J. Delp, 'Forensic analysis of synthetically generated western blot images', *IEEE Access*, (2022).

[30] Francesco Marra, Diego Gragnaniello, Luisa Verdoliva, and Giovanni Poggi, 'Do GANs leave artificial fingerprints?', in *IEEE Conference on Multimedia Information Processing and Retrieval (MIPR)*, (2019).

[31] Alexander Quinn Nichol and Prafulla Dhariwal, 'Improved denoising diffusion probabilistic models', in *International Conference on Machine Learning (ICML)*, (2021).

[32] Sophie J. Nightingale and Hany Farid, 'AI-synthesized faces are indistinguishable from real faces and more trustworthy', *Proceedings of the National Academy of Sciences*, (2022).

[33] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen, 'Hierarchical text-conditional image generation with CLIP latents', *Computing Research Repository (CoRR)*, (2022).

[34] Severi Rissanen, Markus Heinonen, and Arno Solin, 'Generative modelling with inverse heat dissipation', *Computing Research Repository (CoRR)*, (2022).

[35] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer, 'High-resolution image synthesis with latent diffusion models', in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, (2022).

[36] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei, 'ImageNet large scale visual recognition challenge', *International Journal of Computer Vision (IJCV)*, (2015).

[37] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily Denton, Seyed Kamyar Seyed Ghasemipour, Burcu Karagol Ayan, S. Sara Mahdavi, Rapha Gontijo Lopes, Tim Salimans, Jonathan Ho, David J. Fleet, and Mohammad Norouzi, 'Photorealistic text-to-image diffusion models with deep language understanding', *Computing Research Repository (CoRR)*, (2022).

[38] Tim Salimans and Jonathan Ho, 'Progressive distillation for fast sampling of diffusion models', in *International Conference on Learning Representations (ICLR)*, (2022).

[39] Axel Sauer, Kashyap Chitta, Jens Müller, and Andreas Geiger, 'Projected GANs converge faster', in *Advances in Neural Information Processing Systems (NeurIPS)*, (2021).

[40] Katja Schwarz, Yiyi Liao, and Andreas Geiger, 'On the frequency bias of generative models', in *Advances in Neural Information Processing Systems (NeurIPS)*, (2021).

[41] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli, 'Deep unsupervised learning using nonequilibrium thermodynamics', in *International Conference on Machine Learning (ICML)*, (2015).

[42] Jiaming Song, Chenlin Meng, and Stefano Ermon, 'Denoising diffusion implicit models', in *International Conference on Learning Representations (ICLR)*, (2022).

[43] Yang Song and Stefano Ermon, 'Generative modeling by estimating gradients of the data distribution', in *Advances in Neural Information Processing Systems (NeurIPS)*, (2019).

[44] Yang Song and Stefano Ermon, 'Improved techniques for training score-based generative models', in *Advances in Neural Information Processing Systems (NeurIPS)*, (2020).

[45] 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 (ICLR), (2022).
[46] Mingxing Tan and Quoc Le, 'EfficientNet: Rethinking model scaling for convolutional neural networks', in *International Conference on Machine Learning (ICML)*, (2019).
[47] Laurens van der Maaten and Geoffrey Hinton, 'Visualizing data using t-SNE', *Journal of Machine Learning Research (JMLR)*, (2008).
[48] Luisa Verdoliva, 'Media forensics and DeepFakes: An overview', *IEEE Journal of Selected Topics in Signal Processing*, (2020).
[49] Pascal Vincent, Hugo Larochelle, Isabelle Lajoie, Yoshua Bengio, and Pierre-Antoine Manzagol, 'Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion', *Journal of Machine Learning Research (JMLR)*, (2010).
[50] Gregory K. Wallace, 'The JPEG still picture compression standard', *Communications of the ACM*, (1991).
[51] Sheng-Yu Wang, Oliver Wang, Richard Zhang, Andrew Owens, and Alexei A. Efros, 'CNN-generated images are surprisingly easy to spot... for now', in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, (2020).
[52] Zhendong Wang, Huangjie Zheng, Pengcheng He, Weizhu Chen, and Mingyuan Zhou, 'Diffusion-GAN: Training GANs with diffusion', *Computing Research Repository (CoRR)*, (2022).
[53] Zijie J. Wang, Evan Montoya, David Munechika, Haoyang Yang, Benjamin Hoover, and Duen Horng Chau, 'DiffusionDB: A large-scale prompt gallery dataset for text-to-image generative models', *Computing Research Repository (CoRR)*, (2022).
[54] Moritz Wolter, Felix Blanke, Raoul Heese, and Jochen Garcke, 'Wavelet-packets for deepfake image analysis and detection', *Machine Learning*, (2022).
[55] Zhisheng Xiao, Karsten Kreis, and Arash Vahdat, 'Tackling the generative learning trilemma with denoising diffusion GANs', in *International Conference on Learning Representations (ICLR)*, (2022).
[56] Xinsheng Xuan, Bo Peng, Wei Wang, and Jing Dong, 'On the generalization of GAN image forensics', in *Biometric Recognition (CCBR)*, (2019).
[57] Fisher Yu, Ari Seff, Yinda Zhang, Shuran Song, Thomas Funkhouser, and Jianxiong Xiao, 'LSUN: Construction of a large-scale image dataset using deep learning with humans in the loop', *Computing Research Repository (CoRR)*, (2016).
[58] Xu Zhang, Svebor Karaman, and Shih-Fu Chang, 'Detecting and simulating artifacts in GAN fake images', in *IEEE International Workshop on Information Forensics and Security (WIFS)*, (2019).

A Details on the Dataset

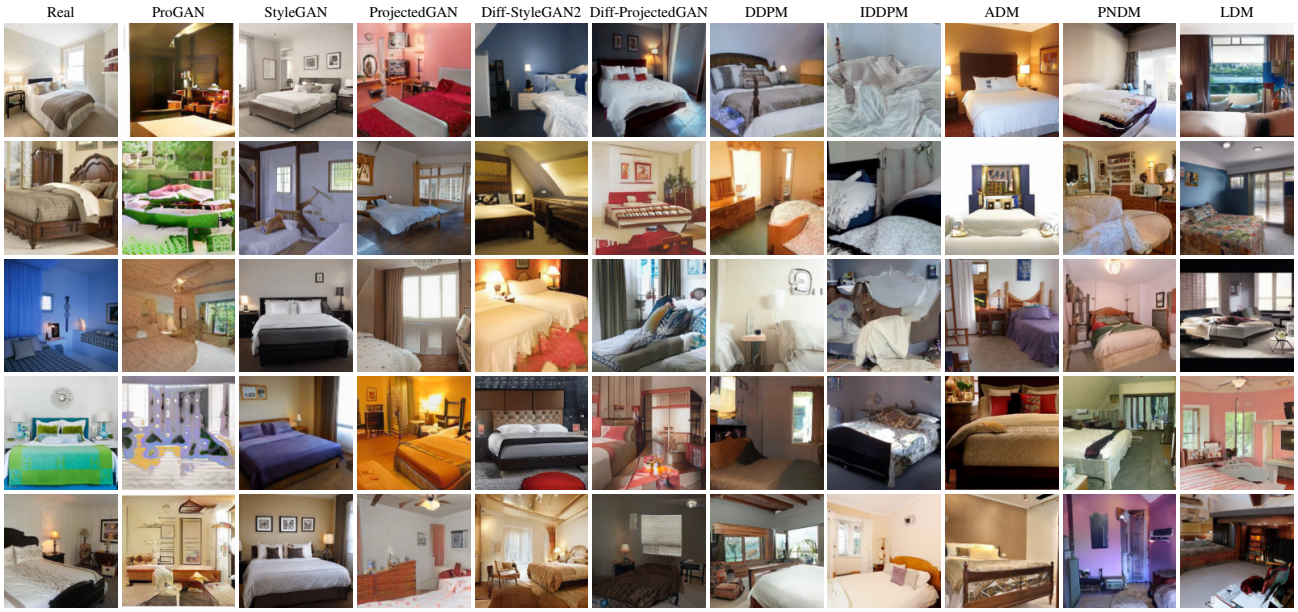


Figure 8. Non-curated examples for real LSUN Bedroom, GAN-generated, and DM-generated images.

LSUN Bedroom [57] We download and extract the lmbd database files using the official repository[4]. The images are center-cropped to 256×256 pixels.

ProGAN [20] We download the first 10k samples from the non-curated collection provided by the authors.[5]

StyleGAN [21] We download the first 10k samples generated with $\psi = 0.5$ from the non-curated collection provided by the authors.[6]

ProjectedGAN [39] We sample 10k images using code and pre-trained models provided by the authors using the default configuration (`-trunc=1.0`).[7]

Diff-StyleGAN2 [52] and Diff-ProjectedGAN [52] We sample 10k images using code and pre-trained models provided by the authors using the default configuration.[8]

DDPM [17], IDDPM [31], and ADM [7] We download the samples provided by the authors of ADM[9] and extract the first 10k samples for each generator. For ADM on LSUN, we select the models trained with dropout.

PNDM [25] We sample 10k images using code and pre-trained model provided by the authors.[10] We specify `-method F-PNDM` and `-sample_speed 20` for LSUN Bedroom and `-sample_speed 10` for LSUN Church, as these are the settings leading to the lowest FID according to Tables 5 and 6 in the original publication.

LDM [35] We sample 10k images using code and pre-trained models provided by the authors using settings from the corresponding table in the repository.[11] For LSUN Church there is an inconsistency between the repository and the paper, we choose 200 DDIM steps (`-c 200`) as reported in the paper.

[4] https://github.com/fyu/lsun
[5] https://github.com/tkarras/progressive_growing_of_gans
[6] https://github.com/NVlabs/stylegan
[7] https://github.com/autonomousvision/projected_gan
[8] https://github.com/Zhendong-Wang/Diffusion-GAN
[9] https://github.com/openai/guided-diffusion
[10] https://github.com/luping-liu/PNDM
[11] https://github.com/CompVis/latent-diffusion

B Detection

B.1 Descriptions of Detectors

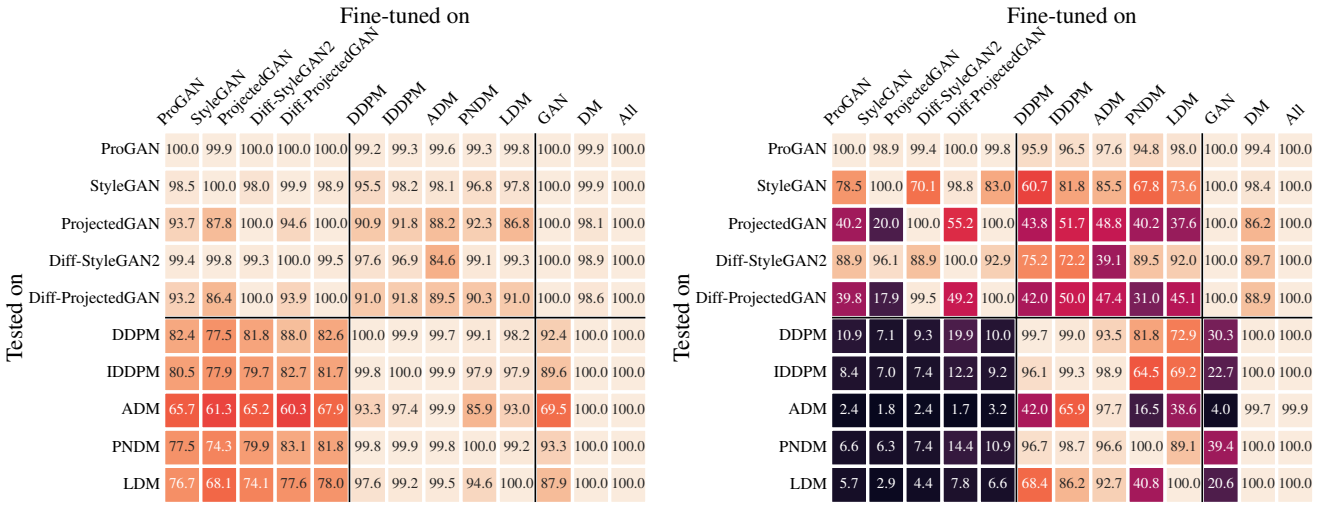
Wang et al. [51] In this influential work, the authors demonstrate that a standard deep convolutional neural network (CNN) trained on data from a single generator performs surprisingly well on unseen images. They train a ResNet-50 [16] on 720k images from 20 LSUN [57] categories (not including Bedroom and Church), equally divided into real images and images generated by ProGAN [20]. The trained binary classifier is able to distinguish real from generated images from a variety of generative models and datasets. The authors further show that extensive data augmentation in the form of random flipping, blurring, and JPEG compression generally improves generalization. Moreover, the authors provide two pre-trained model configurations, Blur+JPEG (0.1) and Blur+JPEG (0.5), where the value in parentheses denotes the probability of blurring and JPEG compression, respectively. They achieve an average precision of 92.6% and 90.8%, respectively. The work suggests that CNN-generated images contain common artifacts (or fingerprints) which make them distinguishable from real images.

Gragnaniello et al. [14] Building upon the architecture and dataset from Wang et al. [51], the authors of this work experiment with different variations to further improve the detection performance in real-world scenarios. The most promising variant, no-down, removes downsampling from the first layer, increasing the average accuracy from 80.71% to 94.42%, at the cost of more trainable parameters. They also train a model with the same architecture on images generated by StyleGAN2 [22] instead of ProGAN, which further improves accuracy to 98.48%.

Mandelli et al. [28] Unlike the other two methods, this work uses an ensemble of five orthogonal CNNs to detect fake images not seen during training. All CNNs are based on the EfficientNet-B4 model [46] but are trained on different datasets. Dataset orthogonality refers to images having different content, different processing (e.g., JPEG compression), or being generated by different GANs. They argue that by having different datasets, each CNN learns to detect different characteristics of real and generated images, improving the overall performance and generalization ability. For each CNN, a score is computed for ≈ 200 random patches extracted from the image. These scores are combined using a novel patch aggregation strategy which assumes that an image is fake if at least one patch is classified as being fake. Finally, the output score is computed by averaging the individual scores of all five CNNs. It should be noted that, besides several GANs, the training dataset also includes samples generated by Song et al. [45], a score-based model.

B.2 Detection Results For Fine-Tuned Classifiers

We repeat the experiment described in Section 5.2 but fine-tune each classifier instead of training it from scratch, the results are shown in Figure 9. For most constellations of training and test data, the results are similar compared to the experiments in Section 5.2, with a tendency towards worse scores for the models trained from scratch.



(a) AUROC (b) Pd@1%

Figure 9. Detection performance for fine-tuned detectors. The columns *GAN*, *DM*, and *All* correspond to models fine-tuned on samples from all GANs, DMs, and both, respectively.

B.3 Feature Space Analysis

The experiments in Section 5.1 and Supplement D demonstrate that current detection methods struggle to reliably identify DM-generated images while detecting GAN-generated images with near-optimal accuracy. We thus hypothesize that GAN-generated images exhibit features that are considerably distinct from those of DM-generated images.

To further investigate this hypothesis we examine the discrepancy of real and fake images in the feature spaces of the detection methods under consideration. More precisely, we employ maximum mean discrepancy (MMD, [15]) to estimate the distance between the distributions of real and generated images in feature space. The 2048-dimensional features are extracted prior to the last fully-connected layer of the detectors by Wang et al. [51] and Gragnaniello et al. [14]. We employ a Gaussian kernel with σ as the median distance between instances in the combined sample as suggested by Gretton et al. [15] and calculate the MMD between the representations of 10k generated images and 10k real images.

We consider three sets of detection methods, namely pre-trained, fine-tuned and trained from scratch. Starting with pre-trained detection methods (top row in Figure 10), we observe that the MMD between representations of DM-generated images and those of real images are considerably lower in comparison to GAN-generated images. With the results from Table 2 in mind, we can conclude that the pre-trained detectors extract features that allow to reliably separate GAN-generated images from real images, while these features are less informative to discriminate DM-generated images. These findings support the experiments conducted in Section 5.1.

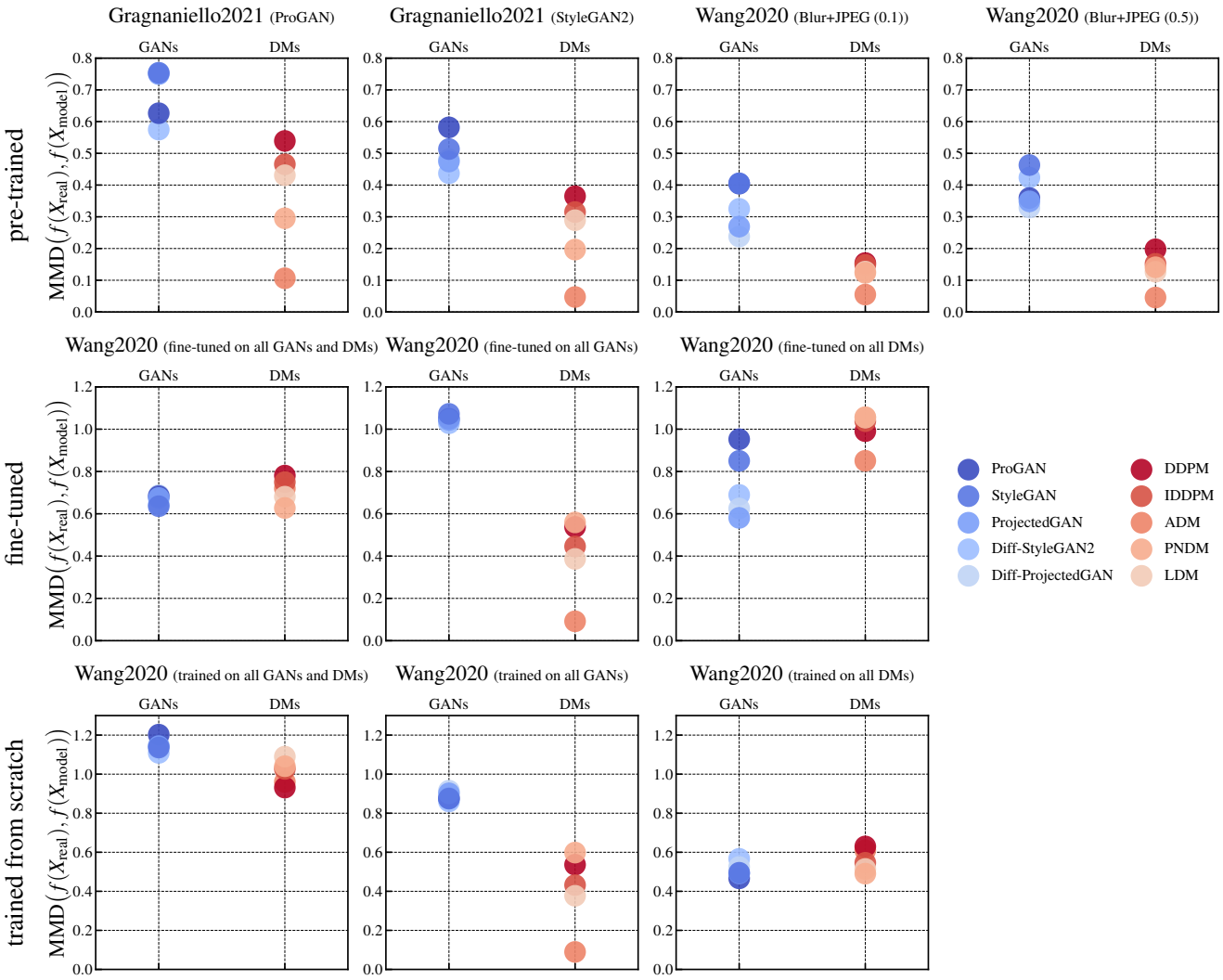


Figure 10. MMD between feature representations of real and generated images on LSUN Bedroom for different detectors (pre-trained, fine-tuned and trained from scratch). The features $f(\cdot)$ correspond to the representation prior to the last fully-connected layer of the respective detection method. For further information on the detectors see Section 5.2 (fine-tuned) and Supplement B.2 (trained from scratch).

We repeat the above experiment for fine-tuned and re-trained detectors from Wang et al. [51] (middle and bottom row in Figure 10). We consider the version Blur+JPEG (0.5) and fine-tune/train on all GAN-generated images, DM-generated images and both (as described in Section 5.2 and Supplement B.2). The results of fine-tuned and re-trained detectors are qualitatively similar: When training/fine-tuning on generated images from both, DMs and GANs, the MMDs in feature space are roughly on par for both model sets. This strengthens the finding that reliable detection is feasible provided the generative model class is known. When fine-tuning/training solely on images from one model class we observe an imbalance: Detectors fine-tuned or trained on images from DMs are able to achieve relatively higher MMDs for GAN-generated images than vice versa. This imbalance is already evident in the experiments in Section 5.2.

The findings give rise to the following hypothesis: GAN- and DM-generated images share some common patterns. However, detectors

trained solely on GAN-generated images focus mostly on GAN-specific (arguably more prominent) patterns, while detectors trained on DM-generated images focus on common patterns. The hypothesis is corroborated by the t-SNE visualizations [47] provided in Figure 11.[12] The detectors trained on DM-generated images evidently learn features/representations which are common to both GANs and DMs, while detectors which saw GAN-generated images during training map GAN- and DM-generated images to distinct representations. Moreover, detectors trained/fine-tuned solely on GAN-generated images lead to representations of real and DM-generated images which are not clearly separated, which is in line with the reduced detection performance.

[12] We used the scikit-learn implementation of t-SNE with the default settings for all 110k images in our dataset (10k images per GAN/DM and 10k real images) and visualize 250 images per class. Details on t-SNE are given at https://scikit-learn.org/stable/modules/generated/sklearn.manifold.TSNE.html.

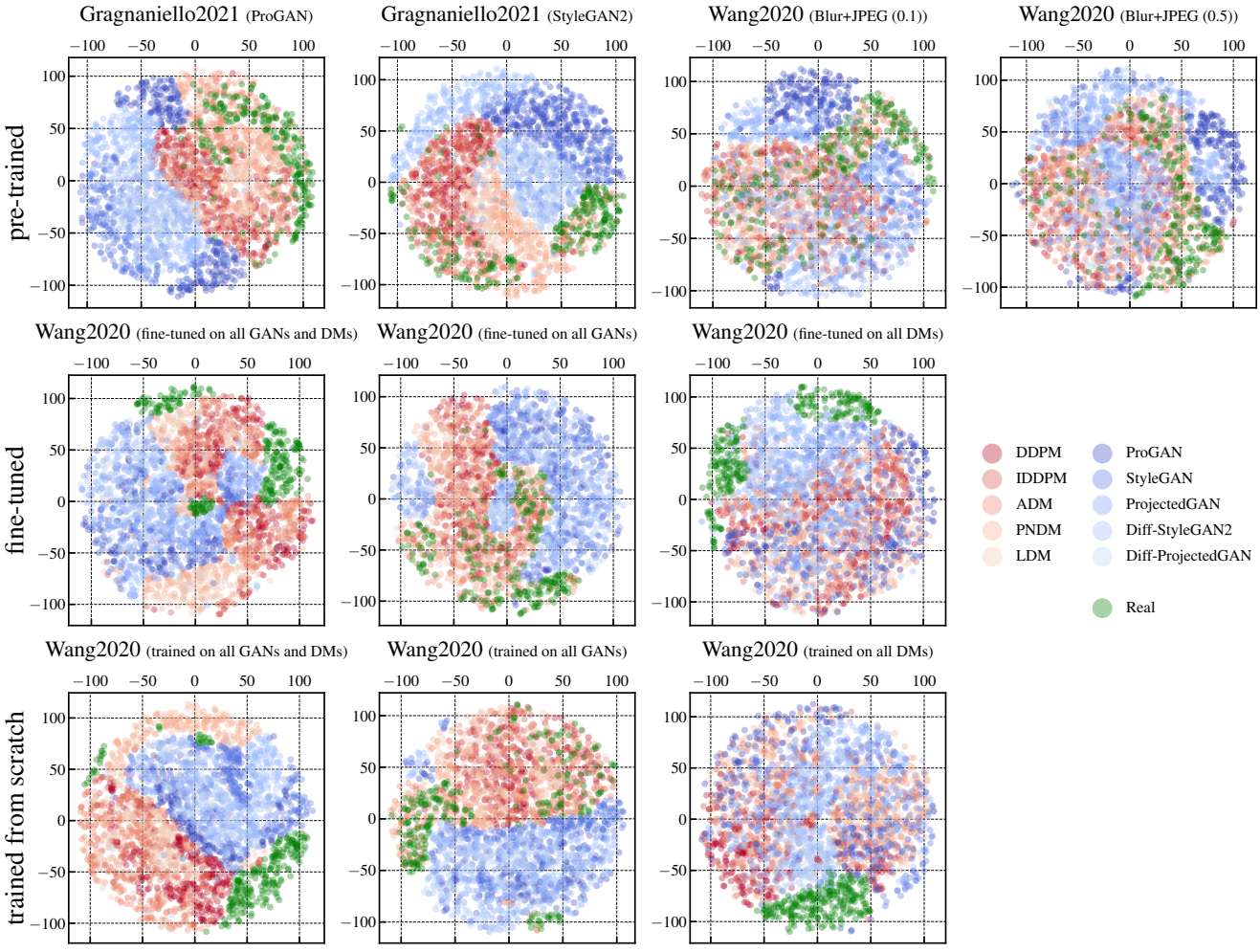


Figure 11. t-SNE visualization of representations of real and generated images on LSUN Bedroom for the detector by Gragnaniello et al. [14] (pre-trained) and Wang et al. [51] (pre-trained, fine-tuned and trained from scratch) in two dimensions. The representations/features $f(\cdot)$ correspond to the representation prior to the last fully-connected layer of the respective detection method and are of dimension 2048. Fine-tuned and re-trained detectors are based on Wang et al. [51] (Blur+JPEG (0.5)). For further information on the detectors see Section 5.2 (trained from scratch) and Supplement B.2 (fine-tuned). Note that a portion of this figure is equivalent to Figure 2 in the main paper.

B.4 Effect of Image Perturbations

In most real-world scenarios, like uploading to social media, images are processed, which is why several previous works consider common image perturbations when evaluating detectors [51, 18, 26, 10]. We follow the protocol of Frank et al. [10] and apply blurring using a Gaussian filter (kernel size sampled from $\{3, 5, 7, 9\}$, cropping with subsequent upsampling (crop factor sampled from $U(5, 20)$), JPEG compression (quality factor sampled from $U(10, 75)$), and Gaussian noising (variance sampled from $U(5, 20)$). Unlike Frank et al. [10], we apply each perturbation with a probability of 100% to study its effect on the detection performance.

Table 3 shows the results for the best-performing detector by Gragnaniello et al. [14] trained on ProGAN images. Note that this detection method employs training augmentation using blurring and JPEG compression. We copy the results on "clean" images from Table 2 as a reference. For GAN-generated images, blurring and cropping have only a very small effect on the detection performance, while compression and the addition of noise cause a stronger deterioration (1% and 3.4% average AUROC decrease, respectively). Overall, we observe that perturbations (except for cropping) have a stronger effect on DM-generated images compared to GAN-generated images. On average, the AUROC decreases by 12.66% for blurring, 14.98% for compression, and 12.18% for noise. We repeat the experiment using the detector by Wang et al. [51] fine-tuned on images from all DMs (see Supplement B.2). As the results in Table 4 show, the effect of blurring and cropping is now almost negligible. The performance drop caused by JPEG compression is also significantly smaller (2.42% average AUROC decrease). While fine-tuning improves the results for these three perturbations, the same does not hold for adding Gaussian noise. The latter could be related to the fact that the image generation process of DMs involves noise.

Table 3. Effect of image perturbations on detection performance. We use the pre-trained detector by Gragnaniello et al. [14] trained on ProGAN images and compute metrics from 10k samples.

AUROC / Pd@5% / Pd@1%	Clean	Blur	Crop	JPEG	Noise
ProGAN	100.0 / 100.0 / 100.0	100.0 / 100.0 / 100.0	100.0 / 100.0 / 100.0	100.0 / 99.8 / 98.6	99.2 / 95.8 / 82.5
StyleGAN	100.0 / 100.0 / 100.0	99.7 / 98.2 / 94.3	100.0 / 100.0 / 99.9	99.0 / 94.2 / 79.9	94.3 / 67.0 / 40.9
ProjectedGAN	100.0 / 99.9 / 99.3	99.2 / 96.3 / 87.6	99.9 / 99.7 / 98.0	98.2 / 88.7 / 71.7	96.6 / 79.9 / 52.8
Diff-StyleGAN2	100.0 / 100.0 / 100.0	100.0 / 100.0 / 100.0	100.0 / 100.0 / 100.0	99.7 / 98.4 / 92.7	96.7 / 79.5 / 56.3
Diff-ProjectedGAN	99.9 / 99.9 / 99.2	99.0 / 94.9 / 84.1	99.9 / 99.7 / 97.0	98.1 / 87.9 / 70.6	96.1 / 77.0 / 51.0
DDPM	96.5 / 79.4 / 39.1	78.5 / 23.7 / 8.1	95.5 / 73.2 / 35.2	81.1 / 21.7 / 6.3	85.1 / 34.0 / 11.2
IDDPM	94.3 / 64.8 / 25.7	75.3 / 18.5 / 5.5	93.5 / 62.0 / 24.6	77.5 / 18.2 / 5.0	80.9 / 25.0 / 7.0
ADM	77.8 / 20.7 / 5.2	66.0 / 10.1 / 3.0	78.3 / 21.4 / 5.5	64.1 / 8.2 / 1.6	64.8 / 8.6 / 1.7
PNDM	91.6 / 52.0 / 16.6	86.7 / 38.8 / 14.3	91.1 / 51.9 / 19.7	76.0 / 15.2 / 3.9	81.2 / 27.4 / 9.3
LDM	96.7 / 79.9 / 42.1	87.1 / 42.8 / 17.1	96.8 / 81.8 / 48.9	83.3 / 25.9 / 8.3	82.2 / 28.6 / 8.8

Table 4. Effect of image perturbations on detection performance. We use the detector by Wang et al. [51] fine-tuned on all DM-generated images and compute metrics from 10k samples.

AUROC / Pd@5% / Pd@1%	Clean	Blur	Crop	JPEG	Noise
DDPM	100.0 / 100.0 / 100.0	100.0 / 100.0 / 99.9	100.0 / 100.0 / 99.9	98.5 / 92.5 / 80.1	69.8 / 22.7 / 11.3
IDDPM	100.0 / 100.0 / 100.0	100.0 / 100.0 / 100.0	100.0 / 100.0 / 100.0	98.1 / 90.3 / 75.8	70.2 / 22.5 / 11.1
ADM	100.0 / 100.0 / 99.7	99.9 / 99.9 / 99.0	100.0 / 100.0 / 99.5	94.5 / 71.8 / 47.1	70.3 / 23.2 / 10.8
PNDM	100.0 / 100.0 / 100.0	100.0 / 99.9 / 99.8	100.0 / 100.0 / 100.0	98.7 / 93.5 / 84.5	74.8 / 29.2 / 14.6
LDM	100.0 / 100.0 / 100.0	100.0 / 100.0 / 100.0	100.0 / 100.0 / 100.0	98.1 / 91.0 / 77.3	71.9 / 25.7 / 12.9

B.5 Fakeness Rating

We evaluate whether DM-generated images exhibit some visual cues used by the detector to distinguish them from real images. Inspired by Wang et al. [51], we rank all images by the model's predictions (higher value means "more fake") and show examples from different percentiles. We consider two detectors, the best-performing pre-trained detector from Gragnaniello et al. [14] trained on ProGAN (Table 12) and the detector from Wang et al. [51] fine-tuned on all DM-generated images (Table 13). Details on the fine-tuning process are given in Supplement B.2. Using the detector from Gragnaniello et al. [14], we make the observation that for most DMs, images which the model assigns a high "fakeness" score contain many pixels which are purely white or black. On the other hand, images considered less fake appear to be more colorful. We were able to reproduce this behavior for other datasets in Figure 14. However, this finding does not hold for the ranking provided by the fine-tuned detector (see Figure 13), which *should* provide more accurate results as its detection performance is significantly higher. Overall, we agree with Wang et al. [51] that there is no strong correlation between the model's predictions and visual quality.

(a) DDPM

(b) IDDPM

(c) ADM

(d) PNDM

(e) LDM

Figure 12. "Fakeness" rating based of DM-generated images (LSUN Bedroom) on predictions from detector by Gragnaniello et al. [14] trained on ProGAN images. Images are ranked by the model's output, i.e., images in the 100th percentile are considered most fake.

(a) DDPM

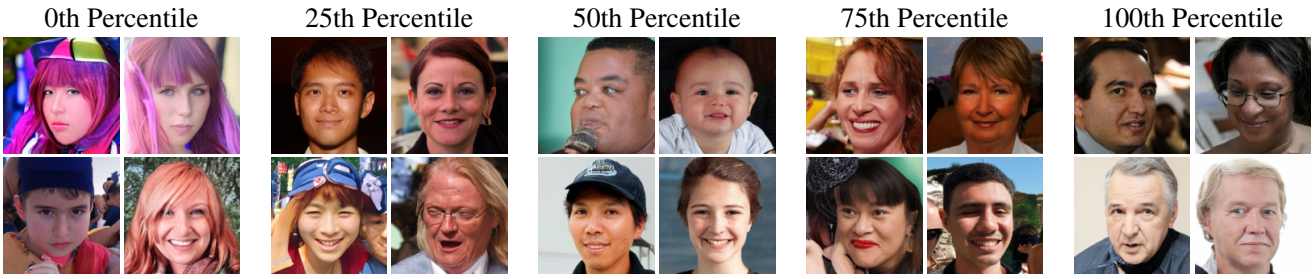
(b) IDDPM

(c) ADM

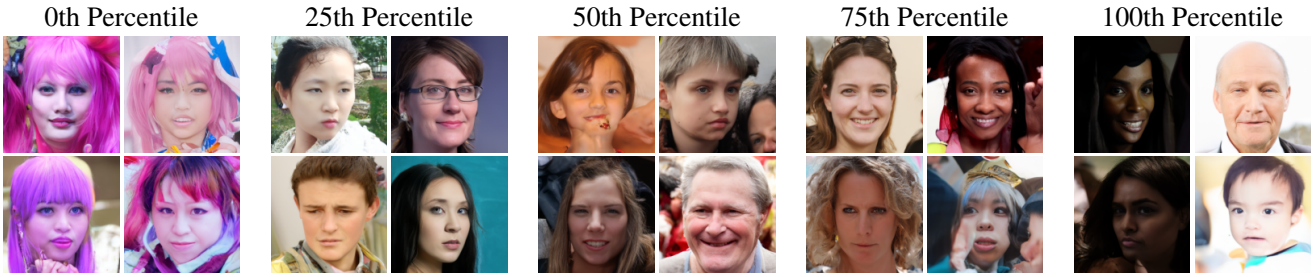
(d) PNDM

(e) LDM

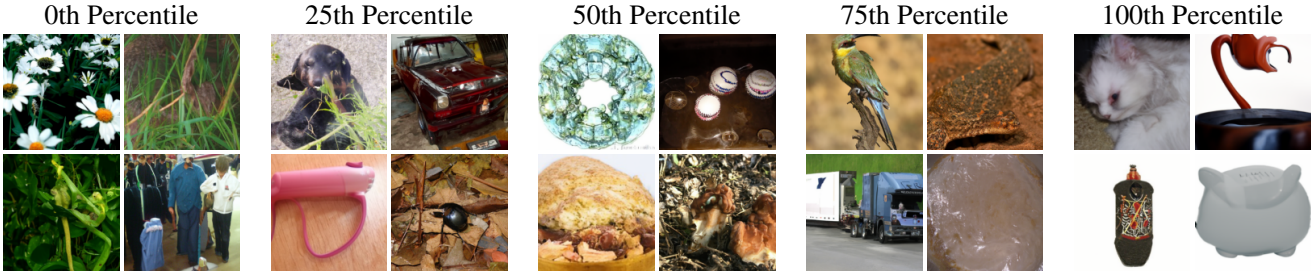
Figure 13. "Fakeness" rating of DM-generated images (LSUN Bedroom) based on predictions from detector by Wang et al. [51] fine-tuned on images from all DMs. Images are ranked by the model's output, i.e., images in the 100th percentile are considered most fake.



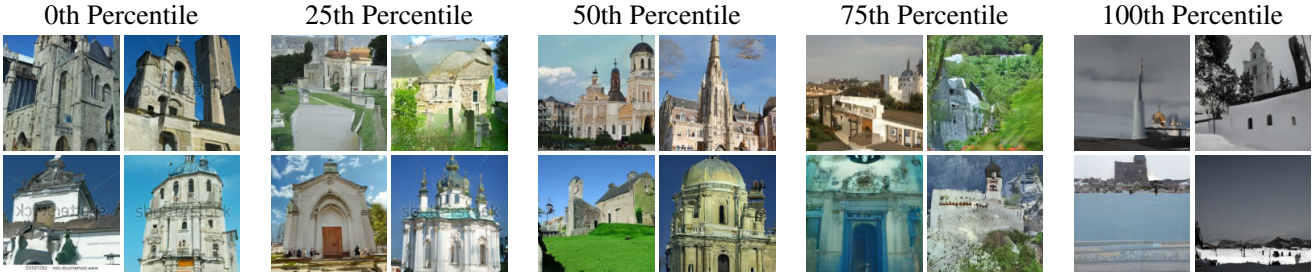
(a) LDM (FFHQ)



(b) ADM' (FFHQ)



(c) ADM (ImageNet)



(d) PNDM (LSUN Church)

Figure 14. "Fakeness" rating of DM-generated images (other datasets) based on predictions from detector by Gragnaniello et al. [14] trained on ProGAN images. Images are ranked by the model's output, i.e., images in the 100th percentile are considered most fake.

C Frequency Analysis

C.1 Frequency Transforms

Discrete Fourier Transform (DFT) The DFT maps a discrete signal to the frequency domain by expressing it as a sum of periodic basis functions. Given a grayscale image I with height H and width W, the two-dimensional DFT (with normalization term omitted) is defined as

$$I_{\text{DFT}}[k,l] = \sum_{x=0}^{H-1}\sum_{y=0}^{W-1} I[x,y]\ \exp^{-2\pi i \frac{x\cdot k}{H}}\ \exp^{-2\pi i \frac{y\cdot l}{W}}, \tag{4}$$

with $k = 0,\dots,H-1$ and $l = 0,\dots,W-1$. For visualization, the zero-frequency component is shifted to the center of the spectrum. Therefore, coefficients towards the edges of the spectrum correspond to higher frequencies.

Discrete Cosine Transform (DCT) The DCT is closely related to the DFT, however it uses real-valued cosine functions as basis functions. It is used in the JPEG compression standard due to its high degree of energy compaction [50], which ensures that a large portion of a signal's energy can be represented using only a few DCT coefficients. The type-II DCT, which the term DCT usually refers to, is given as

$$I_{\text{DCT}}[k,l] = \sum_{x=0}^{H-1}\sum_{y=0}^{W-1} I[x,y] \cos\left[\frac{\pi}{H}\left(x+\frac{1}{2}\right)k_x\right] \cos\left[\frac{\pi}{W}\left(y+\frac{1}{2}\right)k_y\right], \tag{5}$$

again omitting the normalization factor. In the resulting spectrum, the low frequencies are located in the upper left corner, with frequencies increasing along both spatial dimensions.

Reduced Spectrum While the DFT provides a useful visual representation of an image's spectrum, it is less suitable for comparing different spectra quantitatively. Therefore, previous works use the reduced spectrum[13], a one-dimensional representation of the Fourier spectrum [8, 9, 40, 2]. It is obtained by azimuthally averaging over the spectrum in normalized polar coordinates $r \in [0,1]$, $\theta \in [0, 2\pi)$ according to

$$\tilde{S}(r) = \frac{1}{2\pi}\int_0^{2\pi} S(r,\theta)d\theta \quad \text{with} \quad r = \sqrt{\frac{k^2+l^2}{\frac{1}{4}(H^2+W^2)}} \quad \text{and} \quad \theta = \text{atan2}(k,l), \tag{6}$$

with $S[k,l] = |I_{DFT}[k,l]|^2$ being the squared magnitudes of the Fourier coefficients. The maximum frequency is given by the Nyquist frequency $f_{\text{nyq}} = \sqrt{k^2+l^2} = H/\sqrt{2}$ for a square image with $H = W$.

C.2 Logistic Regression Classifier

The experiments in Section 6, in particular Figures 3 and 4, demonstrate that although DMs do not exhibit strong frequency artifacts, their spectrum deviates from that of real images. A natural question is therefore whether these discrepancies can be used to detect DM-generated images more effectively. Following the work of Frank et al. [10], we perform a simple logistic regression on each dataset with different transforms: Pixel (no transform), DFT (and taking the absolute value), and DCT.

We use 20k samples for training, 2k for validation, and 20k for testing, each set equally split between real and fake images. To reduce the number of features, we transform all images to grayscale and take a center crop with dimensions 64×64. Additionally, all features are independently standardized to have zero mean and unit variance. We apply L_2 regularization and identify the optimal regularization weight by performing a grid search over the range $\{10^k \mid k \in \{-4,-3,\dots,4\}\}$.

Table 5. Accuracy of logistic regression on pixels and different transforms. Next to each transform column we report the gain compared to the accuracy on pixels.

	Pixel	DFT		log(DFT)		DCT		log(DCT)	
ProGAN	64.8	74.7	+9.9	72.6	+7.8	65.4	+0.6	**74.9**	+10.1
StyleGAN	91.1	87.4	-3.7	86.2	-4.9	**92.6**	+1.5	86.4	-4.7
ProjectedGAN	90.0	90.8	+0.8	90.3	+0.3	91.0	+1.0	**95.3**	+5.3
Diff-StyleGAN2	92.4	80.3	-12.0	80.5	-11.9	**93.8**	+1.4	87.6	-4.7
Diff-ProjectedGAN	87.4	93.9	+6.5	93.1	+5.7	88.1	+0.6	**97.7**	+10.3
DDPM	51.7	64.2	+12.6	64.2	+12.5	52.4	+0.7	**64.3**	+12.6
IDDPM	51.6	**62.1**	+10.6	61.7	+10.1	51.7	+0.1	61.7	+10.1
ADM	50.1	**54.7**	+4.6	52.3	+2.3	50.1	+0.0	53.8	+3.7
PNDM	52.5	57.0	+4.5	58.2	+5.7	51.1	-1.4	**61.4**	+8.9
LDM	56.6	63.7	+7.1	66.1	+9.5	58.5	+1.9	**73.0**	+16.3

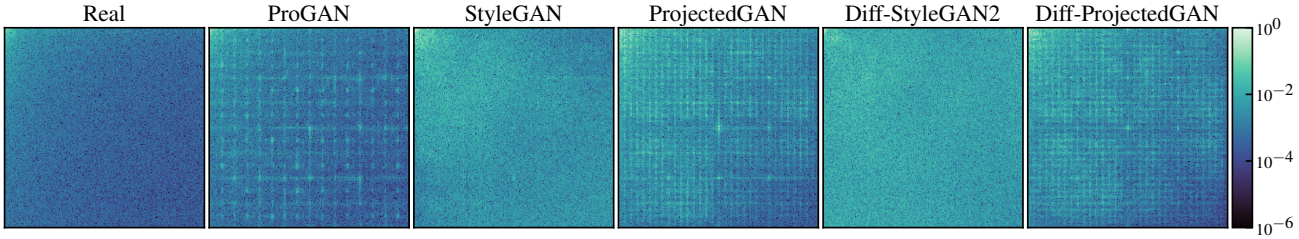
The accuracy of all GANs and DMs in our dataset is given in Table 5. We also report the results for log-scaled DFT and DCT coefficients, as this leads to significant improvements for some generators. For both GANs and DMs, using information from the frequency domain increases

[13] The definition of the reduced spectrum slightly differs between different existing works, we decide to follow that of Schwarz et al. [40].

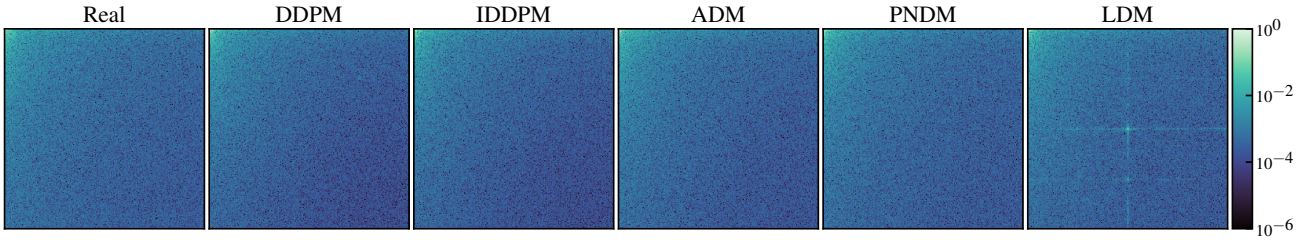
classification accuracy. On average, the performance gain of the best transform compared to no transform is 5.72% and 10.6%, respectively. Although the gain for DMs is more than double that for GANs, the overall maximum accuracy is significantly lower (90.9% for GANs and 63.1% for DMs on average). Therefore, we can not conclude that DMs exhibit stronger discriminative features in the frequency domain compared to GANs. However, these results corroborate the hypothesis that DM-generated images are more difficult to detect.

C.3 DCT Spectra

Figure 15 depicts the DCT spectra of both GANs and DMs. Similar to DFT, images generated by DMs exhibit fewer artifacts, except for LDM.



(a) GANs

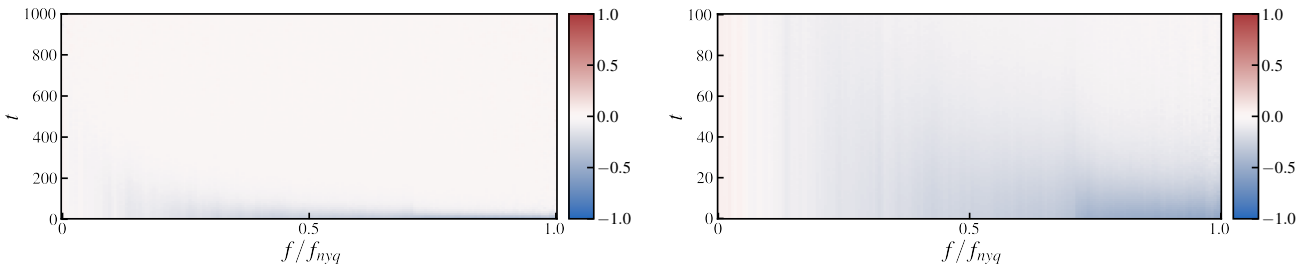


(b) DMs

Figure 15. Mean DCT spectrum of real and generated images. To increase visibility, the color bar is limited to $[10^{-6}, 10^{0}]$, with values lying outside this interval being clipped.

C.4 Spectrum Evolution During the Denoising Process

Analogous to Figure 6, we show the spectrum error evolution during the denoising process in Figure 16. Here, however, the error is computed relative to the spectrum of noised images at the corresponding step during the diffusion process. Similar to the denoising process, the spectra of the diffusion process are averaged over 512 samples. While the relative error is close to zero for a long time during the denoising process, the model fails to accurately reproduce higher frequencies towards $t = 0$. More precisely, too many high-frequency components are removed, which explains why the sweet spot does not seem to be at $t = 0$ but around $t = 10$ (see Figure 6).



(a) $0 \leq t \leq 1000$ (b) $0 \leq t \leq 100$

Figure 16. Evolution of spectral density error throughout the denoising process. The error is computed relatively to the corresponding step in the diffusion process. The colorbar is clipped at -1 and 1. If the model was able to perfectly denoise, the difference would be 0 at all t.

One could think that, by stopping the denoising process early, DM-generated images might be harder to detect. To test this hypothesis, we perform a logistic regression at every $0 \leq t \leq 100$ to distinguish real from increasingly denoised generated images. The model is trained using 512 real and 512 generated samples, from which 20% are used for testing. We select the optimal regularization weight by performing a 5-fold cross-validation over $\{10^k \mid k \in \{-4, -3, \ldots, 4\}\}$. Similar to Section C.2, we compare the performance on pixels and different transforms. The results in Figure 17 do not indicate that around $t = 10$ fake images are less detectable. In Figure 18 it becomes apparent that at this t the images are noticeably noisier, which probably explains the increased accuracy.

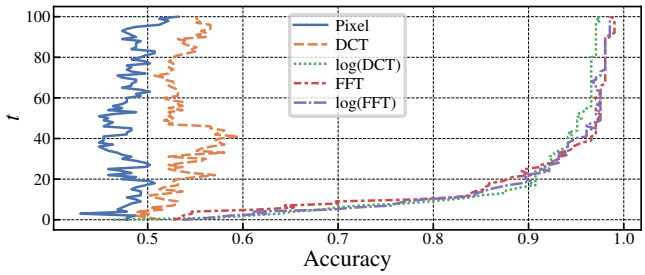


Figure 17. Accuracy of logistic regression during the denoising process. Note that only the last 100 steps of the denoising process are depicted.

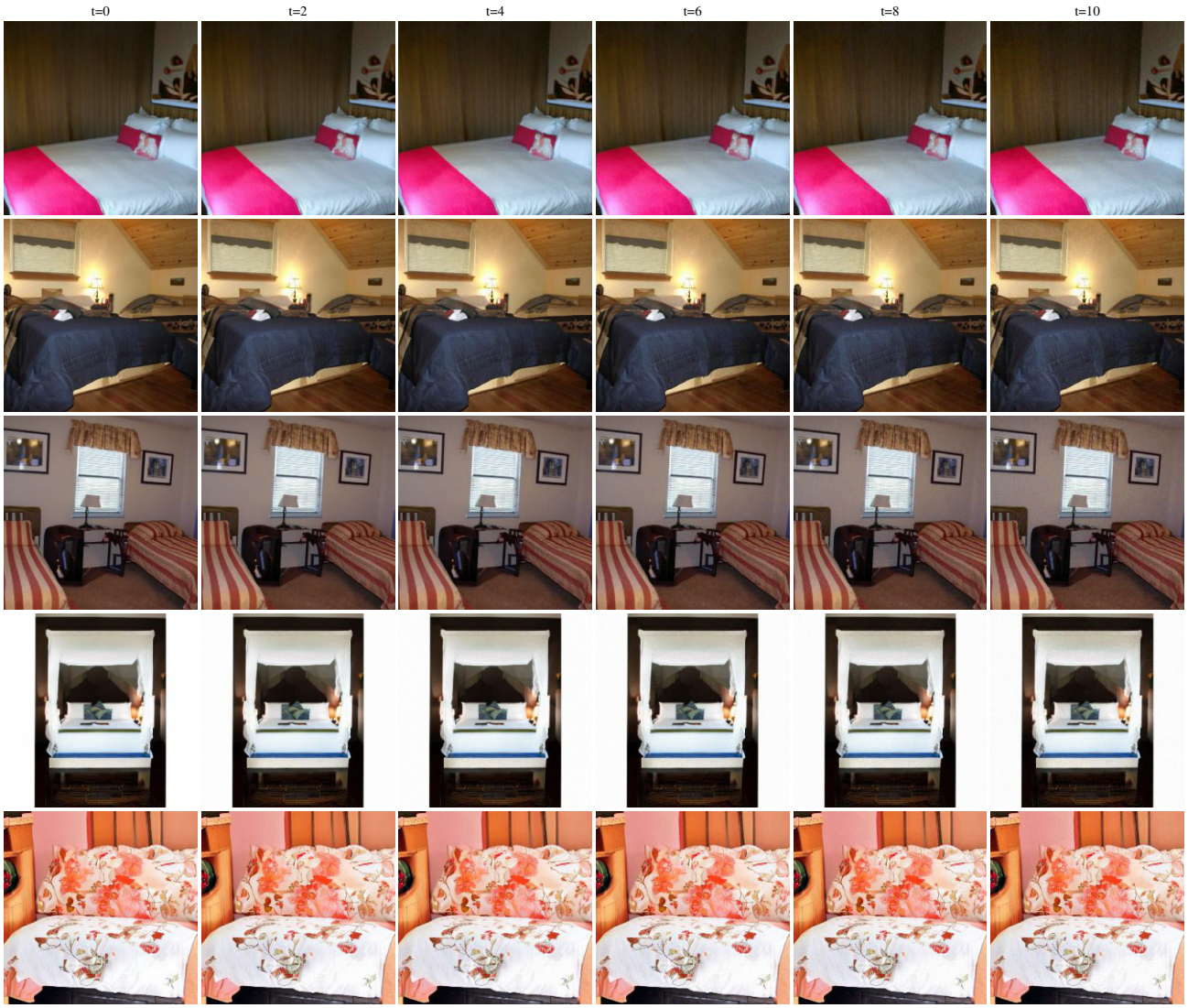


Figure 18. Example images generated by ADM at different t. When zoomed in, the high-frequency noise towards higher t becomes apparent.

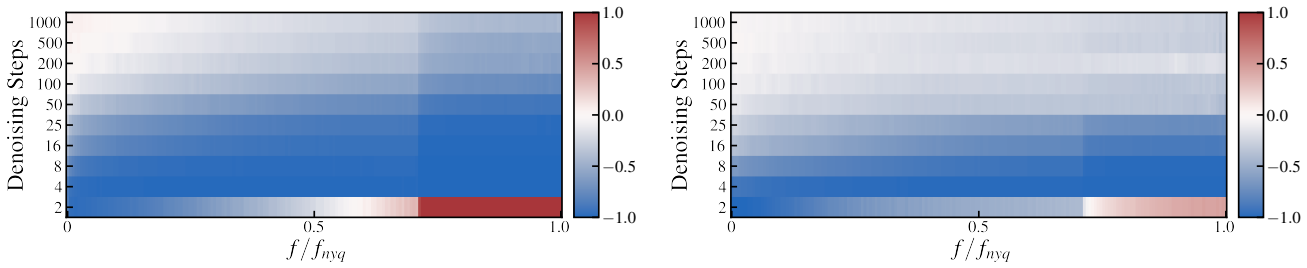
C.5 Effect of Number of Sampling Steps

In this section, we analyze how varying the number of sampling steps affects the spectral properties of generated images. While ADM is usually trained using $T = 1000$ steps, sampling speed can be increased by reducing the number of steps [31]. For ADM on LSUN Bedroom, however, images were sampled using 1000 steps since this led to much better results [7]. The authors also evaluate DDIM [42], an alternative sampling method which allows for sampling with fewer steps.

Using the same experimental settings as in Section 6.2 we analyze the reduced spectra of images sampled using different numbers of steps (Figure 19). With only a few steps, the spectral density is too low across the entire spectrum, resulting in blurry images with little contrast (see

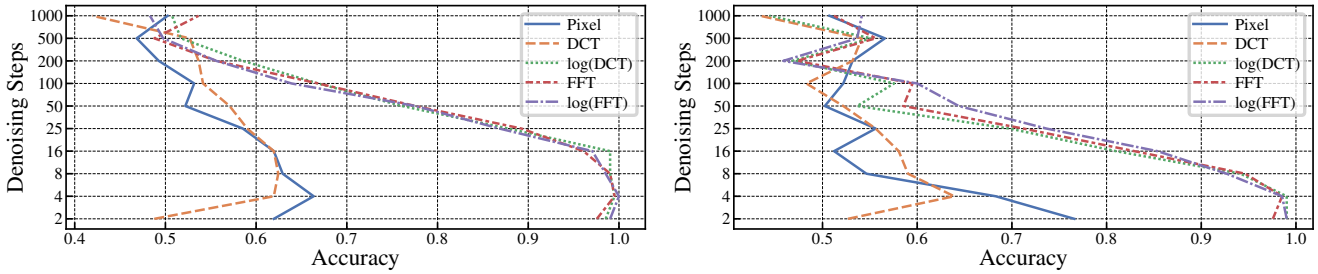
Figure 21). Performing more steps leads to lower errors, with DDIM improving faster than normal sampling. This finding is coherent with previous results, as DDIM achieves better samples at fewer sampling steps [42]. In Figure 20 we also analyze the detectability with different numbers of sampling steps. For both sampling methods, images become easier to detect with fewer sampling steps.

In addition, we repeat the logistic regression experiment from the previous section for samples generated with different numbers of timesteps, the results are shown in Figure 20. We also provide example images in Figure 21. As expected, images become harder to detect with more sampling steps and therefore higher image quality. For DDIM, the accuracy decreases more quickly, which is likely because it was found to generate images of higher quality with fewer sampling steps [42].



(a) Without DDIM (b) With DDIM

Figure 19. Evolution of spectral density error for different numbers of denoising steps. The error is computed relatively to the spectrum of real images. The colorbar is clipped at -1 and 1. Note that the y-axis is not scaled linearly.



(a) Without DDIM (b) With DDIM

Figure 20. Accuracy of logistic regression for different numbers of sampling steps. Note that the y-axis is not scaled linearly.

C.6 Potential Source of the Spectrum Discrepancies

Our analysis in the frequency domain (see Figures 4 in the main paper and 25 in the supplement) suggests that current state-of-the-art DMs do not match the high-frequency content well. To further analyze these findings, we build on insights from the denoising autoencoder (DAE) literature. Roughly speaking, the task DAEs face is conceptually similar to the task of the noise predictor ϵ_θ in DMs at a single time step: denoising a disturbed input at a fixed noise level. Note that while we believe that DMs and DAEs can be conceptually related, the concepts are distinct in several ways: DMs use parameter sharing to perform noise prediction at multiple noise levels to set up the generation as an iterative process. On the other hand, DAEs make use of a latent space to learn suitable representations for reconstruction, classically at a fixed noise level. Nevertheless, we are convinced that it may be useful to take these insights into account.

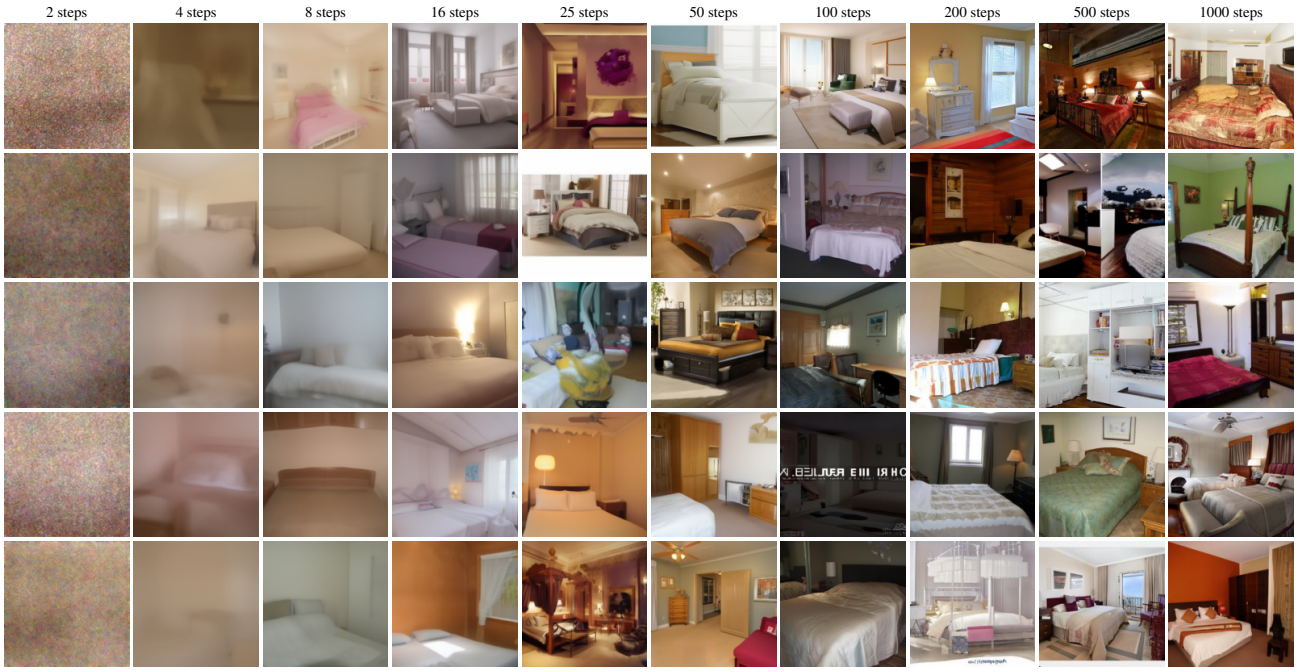
A handy observation from DAEs relates the level of corruption to the learned feature scales: Denoising an image at small noise levels requires to accurately model fine granular details of the image, while coarse/large-scale features can still be recovered at high noise levels (see e.g., [49, 11, 12]). Transferring this insight to DMs, we observe that the training objective guides the reconstruction performance across the different noise levels based on a weighting scheme $w(t)$ that translate to the (relative) importance of certain noise levels throughout the optimization process.

Recall that the objective of many prominent DMs can be stated as a weighted sum of mean squared error (MSE) terms

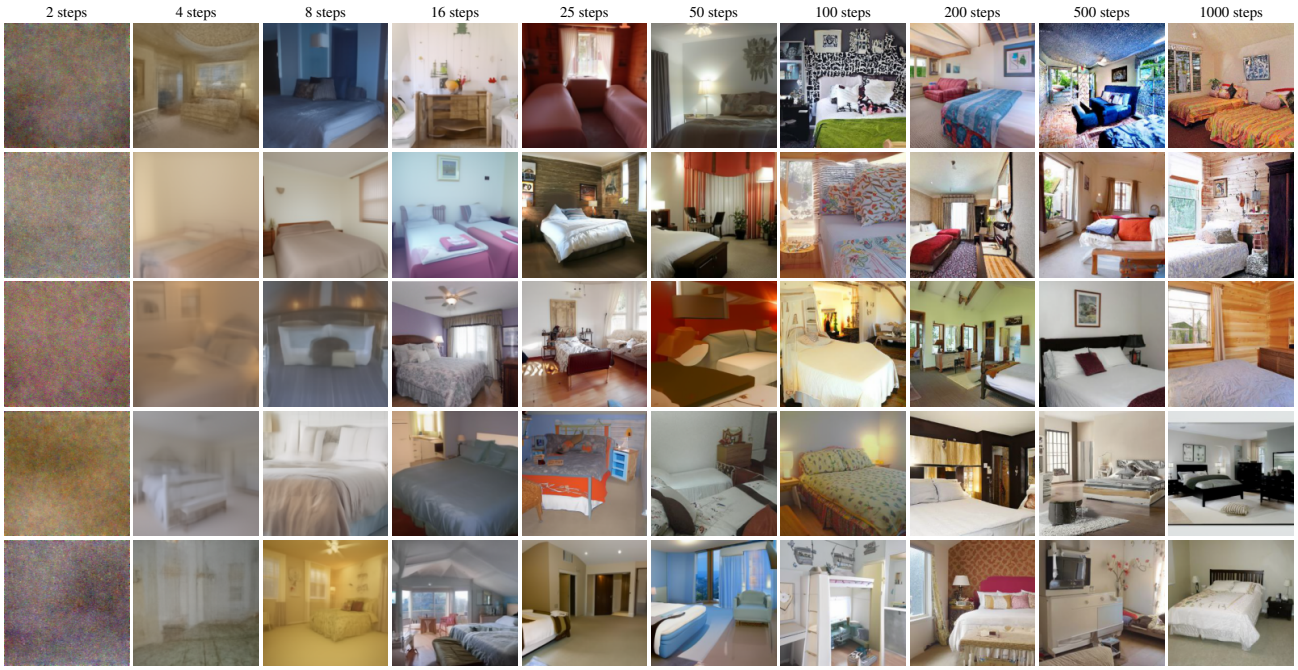
$$L(\theta) = \sum_{t=0}^{T} w(t)\mathbb{E}_{t,\mathbf{x}_0,\epsilon}[\|\epsilon - \epsilon_\theta(\mathbf{x}_t, t)\|^2] \tag{7}$$

usually with $T = 1000$. The theoretically derived variational lower bound L_{vlb}, which would optimize for high likelihood values, corresponds to the weighting scheme (here in the case of $\Sigma(t) = \sigma_t^2\mathbf{I}$)

$$w(t) = \frac{\beta_t^2}{2\sigma_t^2\alpha_t(1-\bar{\alpha}_t)} \tag{8}$$



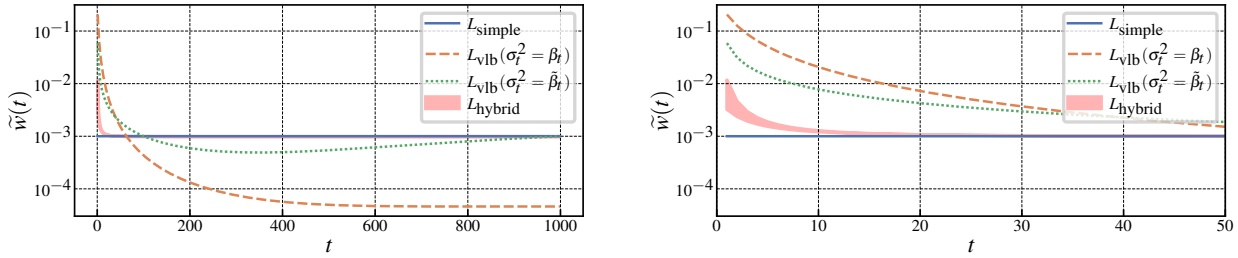
(a) Without DDIM



(b) With DDIM

Figure 21. Example images generated by ADM with different numbers of sampling steps.

with α_t and $\bar{\alpha}_t$ derived from the noise schedule β_t (see Section 3). However, L_{vlb} turns out to be extremely difficult to optimize in practice (see e.g., [7]), arguably due to large influence of the challenging denoising tasks near $t = 0$. To circumvent this issue, L_{simple} was proposed by Ho et al. [17] which corresponds to $w(t) = 1$, and turned out to be stable to optimize and already leads to remarkable perceptual quality. In order to achieve both, perceptual image quality and high likelihood values, Nichol et al. [31] proposed $L_{\text{hybrid}} = L_{\text{simple}} + \lambda L_{\text{vlb}}$ with $\lambda = 0.001$ which increases the influence of low noise levels. The relative importance of reconstruction tasks at different noise levels for the above discussed objectives are depicted in Figure 22.



(a) $0 \leq t \leq 1000$ (b) $0 \leq t \leq 50$

Figure 22. Relative importance of the reconstruction tasks for prominent DM loss functions. We show the relative influence $\widetilde{w}(t) = w(t)/\sum_{t=0}^{T} w(t)$ in Equation 7 for (a) the whole denoising-diffusion process and (b) a close-up on the lowest noise levels. For L_{vlb} we depict both, lower and upper bound for the denoising variance σ_t^2 (given as $\sigma_t^2 = \beta_t$ and $\sigma_t^2 = \tilde{\beta}_t$ as defined by Ho et al. [17]). In the case of L_{hybrid}, used by IDDPM and ADM, we use $\sigma_t^2\mathbf{I}$ for visualization purposes instead of a diagonal covariance matrix and plot $\widetilde{w}(t)$ for the range of admissible variances σ_t^2. All plots assume the linear noise schedule β_t by Ho et al. [17] and $T = 1000$.

To put the above discussion in a nutshell, we hypothesize that the ability of a DM to match the real frequency spectrum is governed by the reconstruction performance at the corresponding noise levels. Importantly, successful error prediction at low noise scales, i.e., near $t = 0$, requires capturing the high-frequency content of an image (which would be the case when (successfully) optimizing L_{vlb}). We deduce that the relative down-weighting of the influence of low noise levels when using L_{simple} or L_{hybrid} (compared to the theoretically derived L_{vlb}) results in the observed mismatch of high frequencies.

The mean reduced spectra of various DMs (Figure 4) support this hypothesis: Both Nichol et al. [31] and Dhariwal et al. [7] train their model with L_{hybrid} (which incorporates a relatively higher weight on low noise levels), and are able to reduce the gap to the real spectrum when compared to the baseline trained with L_{simple} [17]. Clearly, we believe that not only the weighting scheme $w(t)$ accounts for the resulting spectral properties, but more importantly the model's capabilities to successfully predict the low level noise in the first place. From this perspective, the weighting scheme acts as a proxy that encourages the model to focus on specific noise levels.

We conclude that the objectives of DMs are well designed to guide the model to high perceptual quality (or benchmark metrics such as FID), while falling short on providing sufficient information to accurately model the high frequency content of the target images, which would be better captured by a likelihood-based objective like L_{vlb}.

D Additional Results

D.1 Details on the Datasets

To demonstrate the generalization of our findings, we perform classification and frequency analysis on additional data from ADM, PNDM, and LDM. Note that ADM-G-U refers to the two-stage up-sampling stack in which images are generated at a resolution of 64×64 and subsequently up-sampled to 256×256 pixels using a second model [7]. The generated images are obtained according to the instructions given in Section A. Due to the relevance of facial images in the context of deepfakes, we also include two DMs not yet considered, P2 and ADM' [4], trained on FFHQ [21]. ADM' is a smaller version of ADM with 93M instead of more than 500M parameters.[14] P2 is similar to ADM' but features a modified weighting scheme which improves performance by assigning higher weights to diffusion steps where perceptually rich contents are learned [4]. We download checkpoints for both models from the official repository and sample images according to the authors' instructions. Real images from LSUN [57], ImageNet [36], and FFHQ [21] are downloaded from their official sources. Images from LSUN Cat/Horse, FFHQ, and ImageNet are resized and cropped to 256×256 pixels by applying the same pre-processing that was used when preparing the training data for the model they are compared against. For all datasets we collect 10k real and 10k generated images.

Images from Stable Diffusion[15] are generated using the *diffusers* library[16] with default settings. For each version, we generate 10k images using a subset of prompts from DiffusionDB [53]. Since Midjourney[17] is proprietary, we collect 300 images created using the "–v 5" flag from the official Discord server. As real images, we take a subset of 10k images from LAION-Aesthetics V2[18] with aesthetics scores greater than 6.5. For the detection experiments we use the entire images, for computing frequency spectra we take center crops of 256×256 pixels.

For results on other GANs, we refer to the original publications of the detectors [51, 14, 28].

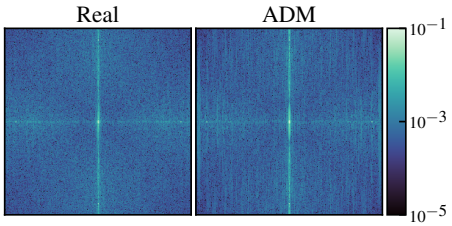
D.2 Frequency Analysis

We analyze the DFT (Figure 23), DCT (Figure 24), and reduced spectra (Figure 25) using a similar process as in Section 6.1. Regarding frequency artifacts, the results are consistent with that from LSUN Bedroom, LDM exhibits grid-like artifacts while ADM and PNDM do not. The spectra of ADM on LSUN Cat and LSUN Horse do contain irregular, vertical structures, which we did not observe for any other model and dataset. However, these are substantially different and not as pronounced as GAN artifacts. The DFT and DCT of (real and generated) FFHQ images clearly deviate from the remaining spectra, which we attribute to the homogeneity of the dataset. Note that LDM and ADM'/P2 were trained on differently processed versions of the real images, which is why we include the spectra of both variants. While we observe the known artifacts for LDM, ADM' and P2 do not contain such patterns.

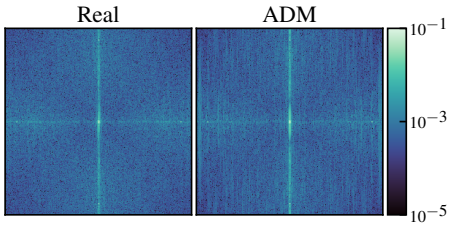
The reduced spectra have largely the same characteristics as for LSUN Bedroom, except for ImageNet. Here we observe an overestimation towards higher frequencies, which is the opposite of what we see for ADM on other datasets. A possible explanation could be that the authors sampled images from LSUN using 1000 and from ImageNet using only 250 steps. We suppose that the amount of spectral discrepancies is highly training dependent.

As discussed in Section 6, the DFT and DCT spectra of images generated by Stable Diffusion exhibit very subtle grid-like artifacts. Note that we exclude images generated by Midjourney from this analysis due to the small number of available samples. The reduced spectra show similarities to those of LDM, with a rise towards the higher end of the spectrum. However, the spectral density of images generated by Stable Diffusion is higher than that of real images throughout the spectrum. It should be noted that these deviations might be caused due to the different data distributions of real and generated images.

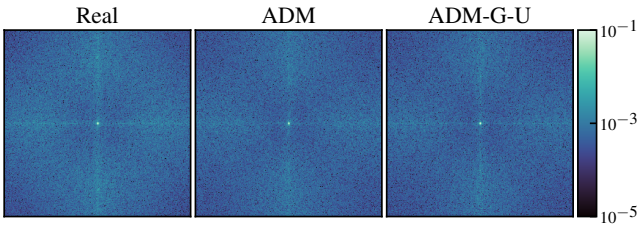
[14] https://github.com/jychoi118/P2-weighting#training-your-models
[15] https://stability.ai/blog/stable-diffusion-public-release
[16] https://huggingface.co/docs/diffusers/index
[17] https://www.midjourney.com
[18] https://laion.ai/blog/laion-aesthetics/



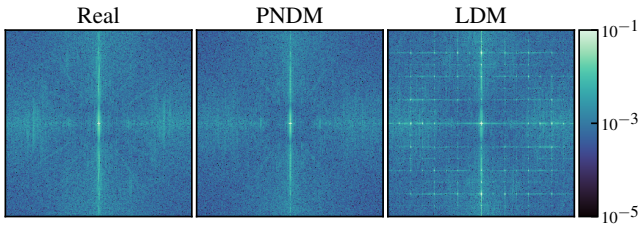
(a) LSUN Cat



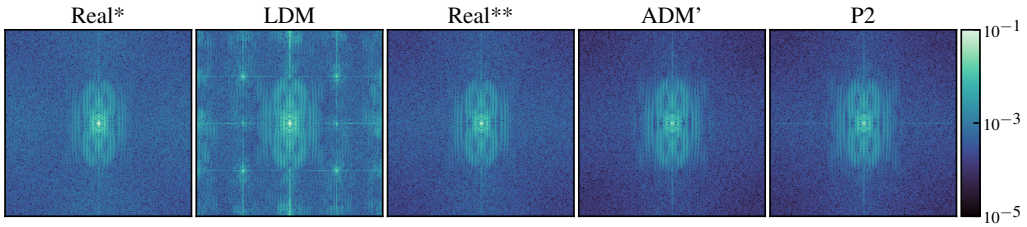
(b) LSUN Horse



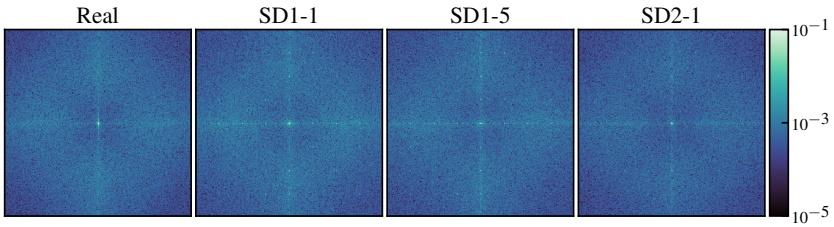
(c) ImageNet



(d) LSUN Church

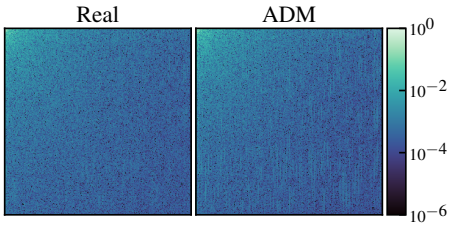


(e) FFHQ (*: pre-processing according to LDM [35], **: pre-processing according to P2 [4])

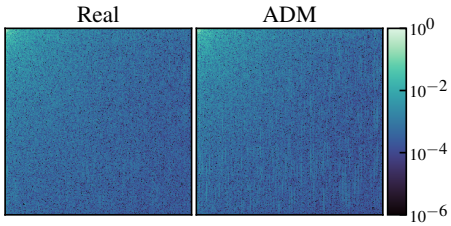


(f) Stable Diffusion

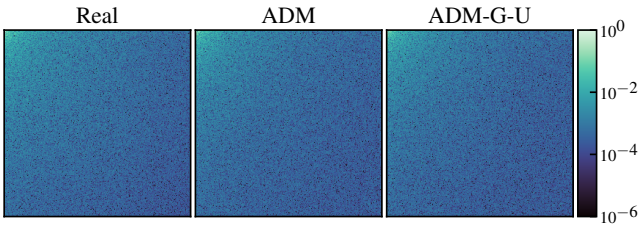
Figure 23. Mean DFT spectrum of real and generated images from additional datasets. To increase visibility, the color bar is limited to $[10^{-5}, 10^{-1}]$, with values lying outside this interval being clipped. Note that parts of (f) are equivalent to Figure 5 and only added here for completeness.



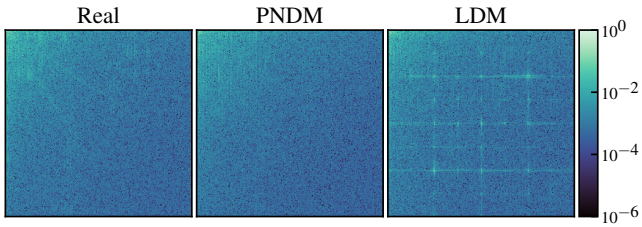
(a) LSUN Cat



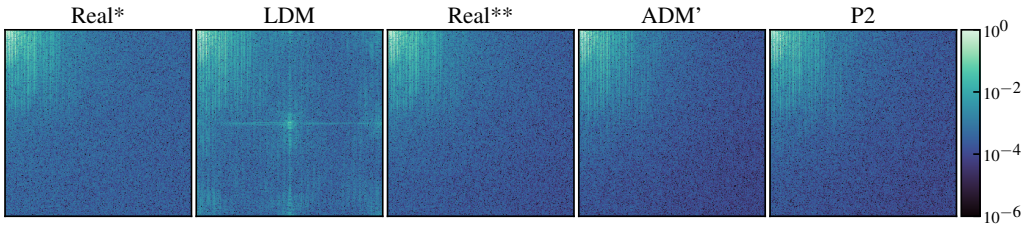
(b) LSUN Horse



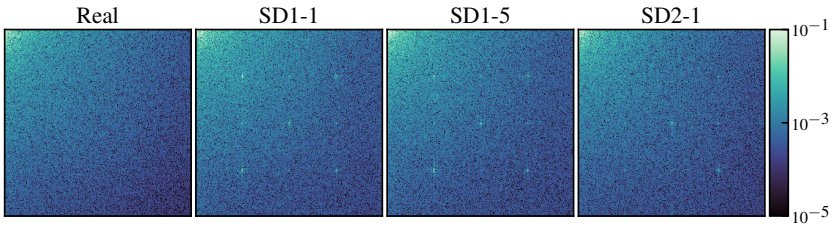
(c) ImageNet



(d) LSUN Church

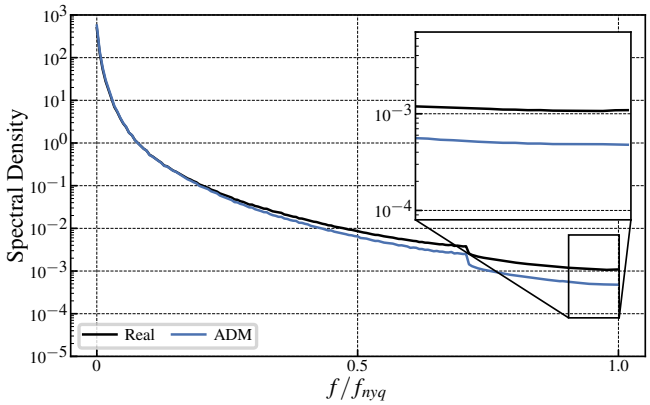


(e) FFHQ (*: pre-processing according to LDM [35], **: pre-processing according to P2 [4])

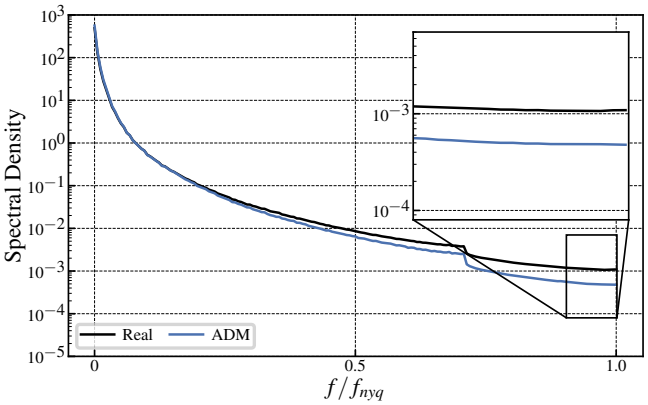


(f) Stable Diffusion

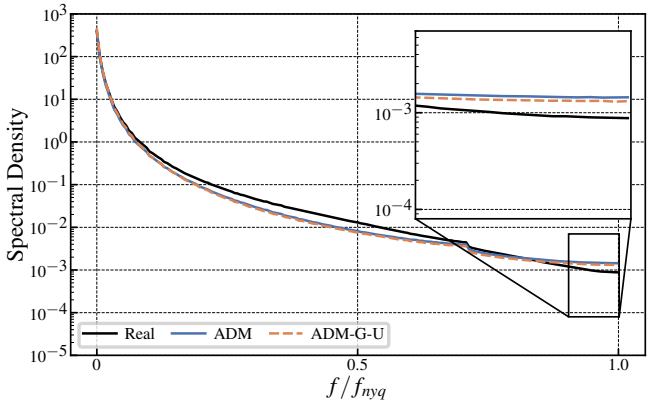
Figure 24. Mean DCT spectrum of real and generated images from additional datasets. To increase visibility, the color bar is limited to $[10^{-6}, 10^{0}]$, with values lying outside this interval being clipped.



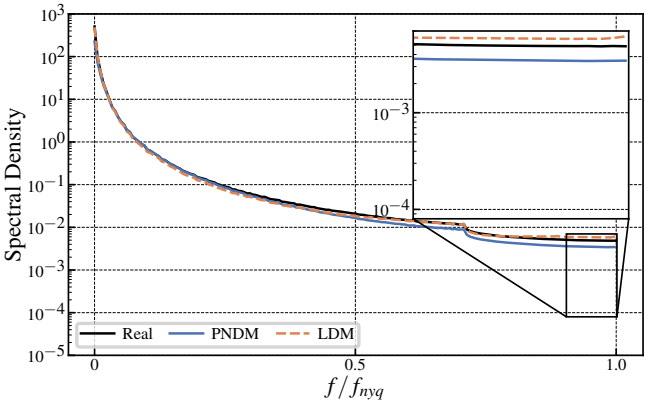
(a) LSUN Cat



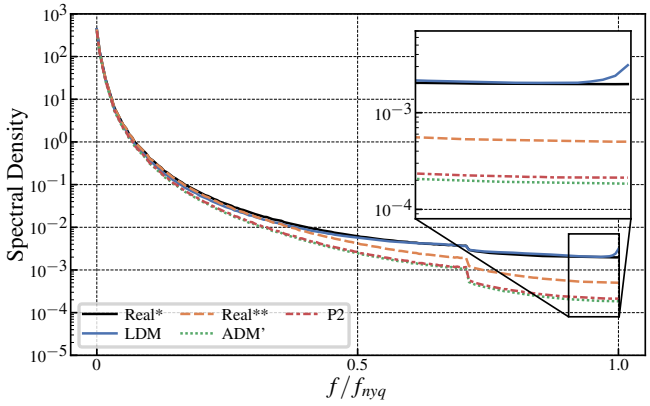
(b) LSUN Horse



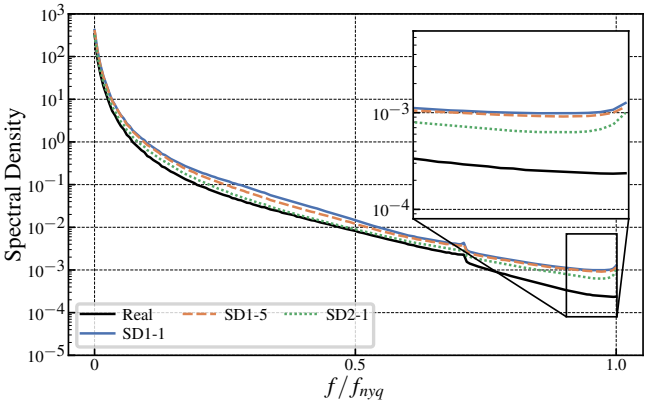
(c) ImageNet



(d) LSUN Church



(e) FFHQ (*: pre-processing according to LDM [35], **: pre-processing according to P2[4])



(f) Stable Diffusion

Figure 25. Mean reduced spectrum of real and generated images from additional datasets. The part of the spectrum where GAN-characteristic discrepancies occur is magnified.