# Community Forensics: Using Thousands of Generators to Train Fake Image Detectors

Jeongsoo Park Andrew Owens
University of Michigan

## Abstract

*One of the key challenges of detecting AI-generated images is spotting images that have been created by previously unseen generative models. We argue that the limited diversity of the training data is a major obstacle to addressing this problem, and we propose a new dataset that is significantly larger and more diverse than prior works. As part of creating this dataset, we systematically download thousands of text-to-image latent diffusion models and sample images from them. We also collect images from dozens of popular open source and commercial models. The resulting dataset contains 2.7M images that have been sampled from 4803 different models. These images collectively capture a wide range of scene content, generator architectures, and image processing settings. Using this dataset, we study the generalization abilities of fake image detectors. Our experiments suggest that detection performance improves as the number of models in the training set increases, even when these models have similar architectures. We also find that increasing the diversity of the models improves detection performance, and that our trained detectors generalize better than those trained on other datasets. The dataset can be found in* https://jespark.net/projects/2024/community_forensics

## 1. Introduction

Our ability to automatically generate realistic images is quickly outpacing our ability to detect them, potentially leading to a state of affairs in which neither humans nor machines can reliably tell real from fake. While the field of image forensics has been developing methods to address this problem, existing fake image detectors still struggle with generalization. These methods often excel at detecting images from generators that were present in their training sets, but fail when given images sampled from unseen models [90, 105, 106, 129].

A core challenge is dealing with the large amounts of variation between models. Each generator has a potentially unique combination of the architecture, loss function, and training distribution. Even seemingly minor differences in low-level image processing details, such as the ways that training images are resized or compressed, can strongly influence detection accuracy [129]. As a result of these model-specific idiosyncrasies, a generator's images may evade detection, even when images from architecturally similar models exist in the training set. This issue has been exacerbated by the thousands of open source models that are now available online, many of which extend pretrained base models in complex ways.

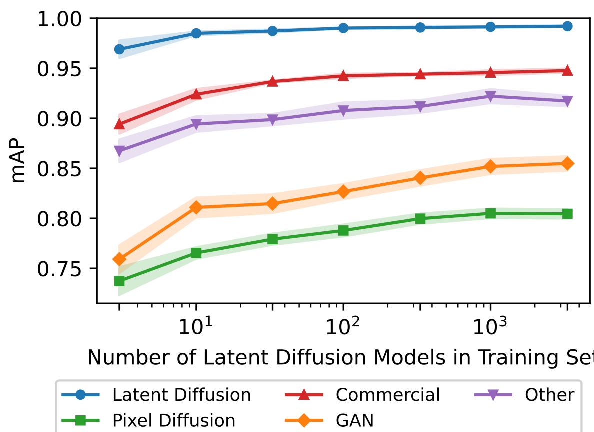
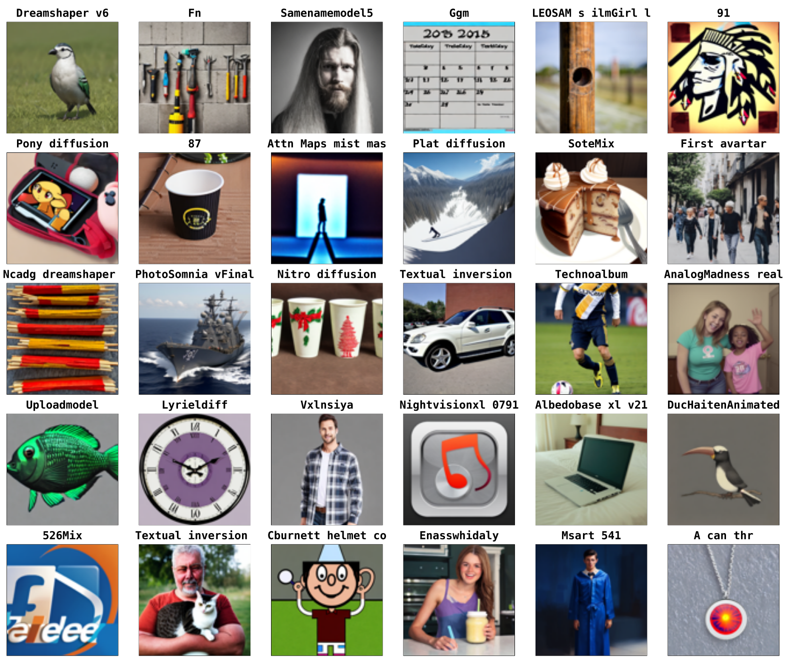



Figure 1. **Performance *vs.* model diversity.** We use images sampled from different numbers of open source latent diffusion models in our *Community Forensics* dataset to train fake image detectors (shown in Fig. 2a). We fix the total number of images and only vary the number of models. The detector's performance increases across all generator types as we train from more models, even though these added models are entirely latent diffusion. This improvement is largest for test images from out-of-distribution generators, such as pixel-based diffusion models or GANs.

We hypothesize that the lack of diversity in training datasets is a major source of these shortcomings. Although today's datasets often contain millions of fake images, they come from a relatively small number of generators. As a result, this data fails to capture many sources of variation that one might encounter in the wild. These limitations also make it challenging to accurately benchmark performance, since it is easy for cues that work well on one set of generators to fail on others.




(a) Systematically collected diffusion models (4763 models)


(b) Manually chosen open source models (19 models)


(c) Commercial models (11 models)

Figure 2. **The *Community Forensics* dataset.** Our dataset contains images sampled from three types of generative models. (a) We systematically download open-source latent diffusion models from a model-sharing community [34, 127]. (b) We select popular open source generators with a variety of architectures and training procedures. (c) We sample from both closed and open state-of-the-art commercial models. We present example images and their corresponding model names.

To address these problems, we propose *Community Forensics*, a dataset that is significantly more diverse and comprehensive than those in prior works (Fig. 2). Our dataset contains images generated by: (a) thousands of systematically downloaded open-source latent diffusion models, (b) hand-selected open source models with various architectures, and (c) state-of-the-art commercial models.

To acquire large numbers of models, we sample images from thousands of text-to-image diffusion models hosted on a popular model-sharing website, Hugging Face [34]. We exploit the fact that these models use a common programming library [127] and thus can be sampled in a standardized way. A large fraction of them are extensions of Stable Diffusion [104], but collectively capture a variety of common model variations, such as in the architecture, image processing, and image content. We also sample images from many other open source models, including GANs [37], autoregressive models [39], and consistency models [80, 118]. To ensure sufficient diversity in the image content of the generated images, we sample generators using captions sourced from a mix of several existing datasets whenever possible.

Our dataset contains 4803 distinct models, roughly 34$\times$ more than the most extensive previous dataset that sample images from generative models [5, 7, 15, 32, 43, 90, 129, 137], and covers a variety of recent model designs (Fig. 2).

We use this dataset to study generalization in the generated image detection problem. Our experiments confirm that increasing the diversity of data improves generalization, consistent with previous findings in other areas of computer vision [48, 111, 112]. In contrast to other work, our dataset aims to increase diversity via *number of generators*, an underexplored axis of diversity. Through experiments, we find:

- *Classifiers trained on our dataset obtain strong performance*, both on our newly proposed evaluations and on multiple previously-proposed benchmarks.
- *Adding more generative models improves generatlization*. Fig. 1 demonstrates the performance of fake image detection when trained on samples from varying numbers of diffusion models. Notably, the performance improves as more models are added, even across different architectures.
- *Including diverse generative model architectures significantly improves results*, since classifiers do not fully generalize between generator architectures.
- *Standard classifiers perform well.* In contrast to observations from recent work, we find that end-to-end training of classifiers based on CNNs or ViTs generalizes well, with qualitatively similar to that of other recognition problems.

## 2. Related work

**Datasets for detecting generated images.** A number of datasets have been proposed for specifically detecting "deepfake" images containing manipulated faces [25, 63, 67, 71, 105, 106, 138]. Rather than focusing on face manipulation, we address creating general-purpose methods that can detect images that have been directly produced by generative models. Wang *et al*. [129] proposed a widely-used dataset of CNN-generated images, mixing images from GANs [8, 12, 58, 59, 94, 136] with other models [9, 10, 18, 70, 106]. This work showed that forensics models generalize between generative models, providing motivation for training on large datasets of diverse generators. However, their classifier was trained on images from a single GAN and was highly sensitive to data augmentation parameters, and more recent work shows that it does not generalize to newer models [14, 90]. Ojha *et al*. [90] introduced a dataset of recent diffusion models and found that training a linear classifier on CLIP features [100] extracted from ProGAN-generated images performed well. Cozzolino *et al*. [15] extend this work by studying the performance of CLIP-based detectors on various generative models and datasets. Epstein *et al*. [32] simulated detecting fake images in an online way by training a detector up to a certain year and testing it on generators released after that year. Zhu *et al*. [137] collected 1.4M generated images from 8 different generators. These datasets, however, only consider a handful of models (less than 20 each), limiting the generalization of their detectors. Asnani *et al*. proposed RED116 [5], collecting 116K images from 116 generative models for predicting the model hyperparameters of a given generated image. Guo *et al*. [43] extends RED116 to 140 generative models with 140K images total. We improve upon these works by collecting much more diverse generative models to improve the performance and generalization of the detector. In concurrent work, Hong *et al*. [49] acquires user-created images from Midjourney and CivitAI. This strategy is complementary to ours: while it aims to collect in-the-wild fake images, its distribution is centered on images that users share, and the models are not necessarily identifiable, making it challenging to rigorously analyze the dataset's contents and to interpret experiments conducted on it.

**Fingerprint-based image forensics methods.** Classic work on image forensics relied on methods based on image statistics [99] and physical constraints [56], rather than learning. A number of datasets have been created for detecting images that have been manipulated using traditional methods, such as with photo editors [21, 26, 53, 64, 87]. Recent works focus on detecting synthetic images by inspecting the generator fingerprints. Zhang *et al*. [135] and Marra *et al*. [81] proposed identifying the spatial fingerprints left by the generator to detect synthetic images. Others focus on spectral anomalies to detect synthetic images. Durall *et al*. [29] and Dzanic *et al*. [30] identified that CNN-generated images fail to reproduce certain spectral properties of real images. Corvi *et al*. [14] study the frequency fingerprints of the generated images and analyzes the cross-architecture generalization of the detector. Bammey [7] uses high-frequency artifacts to detect generated images. However, these approaches may be brittle since the artifacts they rely on can be eliminated by post-processing [15]. We instead approach this problem in a data-driven manner, scaling the number of models, images, and architectures. Recent work has created ensembles of fake image classifiers [50]. In parallel, researchers have detected text generated by language models using supervised learning and heuristics [6, 35, 55, 65, 85, 107, 117, 125], which closely resemble those in visual forensics. However, no existing techniques that we are aware of aim to collect comprehensive datasets of community-created generators.

**Out-of-distribution generalization.** Our work is related to the out-of-distribution recognition problem as it involves generalizing to unseen generators and image processing pipelines. A variety of approaches have been proposed for this problem, based on likelihood ratios [72, 102, 132], self-supervision [47, 86, 113, 128], internal model statistics [46, 108], temperature scaling [3, 73], and via energy-based models [28, 31, 75]. Works by Schuhmann *et al*. [112] and Hendrycks *et al*. [48] show that diverse training data and data augmentation are important to improving the robustness to out-of-distribution samples. Our results are in line with these conclusions, as we find that a diverse set of generative models and stronger augmentations improve generalization.

## 3. The *Community Forensics* Dataset

To support our goal of studying generalization in generated image detection, we collect a dataset of images sampled from a wide range of models (Fig. 2). Our dataset consists of: (a) a large and systematically collected set of "in-the-wild" text-to-image latent diffusion models obtained from a model-sharing website, (b) hand-selected models from other open source architectures, and (c) closed and open state-of-the-art commercial models. We also pair these generated images with real images from other datasets. We preserve the original image format where possible, without any additional compression or resizing. This is to mitigate potential bias and performance degradation in out-of-distribution settings due to unwanted artifacts [40, 42, 101]. Our dataset contains significantly more models than previous works (Tab. 1) and spans a wider range of architectures, processing pipelines, and semantic content.

### 3.1. Systematically collecting generative models

We perform our systematic collection using publicly available, open source[1] models that use the Hugging Face `diffusers` library [34, 127] because: 1) it is a popular library for creating text-to-image models and is widely used by hobbyists, 2) thousands of such models are publicly indexed, and 3) it provides a standard interface by which we can sample images. We process the models in the order of popularity, as indicated by the number of downloads. Our

[1]We use "open source" to describe models with public weights and code, even if they may be closed in some aspect (*e.g*., private training data).

| Dataset | Models | Images | Architectures | Training setup |
|---|---|---|---|---|
| Wang *et al*. [129] | 11 | 362K | GAN, Perceptual, Deepfake, ... | ProGAN [58] *vs*. LSUN [133] |
| Ojha *et al*. [90] | 4* | 10K* | GAN, Perceptual, Diffusion, ... | ProGAN [58] *vs*. LSUN [133] |
| Epstein *et al*. [32] | 14 | 570K | Diffusion | Diffusion *vs*. LAION [112] |
| Cozzolino *et al*. [15] | 18 | 26K | Diffusion | LDM [104] *vs*. MS-COCO [74] |
| Synthbuster [7] | 9 | 10K | Diffusion | Diffusion *vs*. Dresden [36] |
| GenImage [137] | 8 | 1.4M | Diffusion, GAN | Diffusion, GAN *vs*. ImageNet [23] |
| RED116 [5] | 116 | 116K | GAN, VAE, Autoregressive, ... | Many *vs*. Many |
| RED140 [43] | 140 | 140K | Diffusion, GAN, VAE, ... | Many *vs*. Many |
| Ours | 4803 | 2.7M | Diffusion, GAN, Autoregressive, ... | Many *vs*. Many |

Table 1. **Comparison with existing forensics datasets.** We compare the size of the dataset with existing datasets containing identifiable generative models. We only count the number of generated images. Our dataset contains significantly more generative models than prior works. $*$: Only counting the unique evaluation set by Ojha *et al*. [90] as their dataset is based on Wang *et al*. [129].

pipeline downloads each model and extracts relevant hyperparameters (e.g., number of diffusion steps and guidance scale), sampling pipeline configurations, and metadata, from both the model cards on the model-sharing webpage and the metadata retrieved by the `diffusers` library [34, 127]. We sample images using a distribution of text prompts obtained from various real datasets (Sec. 3.3). Since experiments suggest diminishing returns for repeatedly sampling from any given model, we sample a few hundred images from each one. We sample 4763 models with approximately 403 images each, for a total of 1.9M images from this process.

While the lack of documentation in each model and the scale of data collection make it challenging to exactly characterize the model designs in this set, they appear to be entirely (or almost entirely) based on latent diffusion. More specifically, we categorize models as being based on *latent diffusion* if they perform a denoising process on a latent representation.[2] Based on this criterion and the self-reported tags, all models in our systematically collected set appear to be based on latent diffusion. While pixel-based diffusion models also use the `diffusers` library (e.g., DeepFloyd [22]), they were incompatible with our automated generation pipeline. We record such incompatible models and manually sample them to either construct an out-of-distribution test set (Sec. 3.4), or as manually-chosen models for training data (Sec. 3.2).

We show examples of the sampled images in Fig. 2. In Appendix D, we provide examples of models and information from their project pages. These models generate a variety of images, with various types of semantic content and preprocessing. For example, a large fraction of these models adapt variations of a popular pretrained latent diffusion model, Stable Diffusion [104], to different downstream applications, and use a number of adaptation strategies (e.g., using LoRA [51]). We provide the model metadata with each image to enable other possible forensics applications. We discuss these in Appendix B and provide information about image and model licenses.

[2]We note that this definition includes latent consistency models [79].

### 3.2. Collecting images from other architectures

**Images from manually chosen models.** To ensure that our dataset contains a broader range of models, we manually select 19 models from public repositories and sample an average of 40,738 images per model, resulting in 774K images total. We note that this number is itself on par with (or more than) many prior datasets with identifiable generative models [7, 15, 32, 90, 129, 137]. We include several GANs (e.g., StyleGANs [60–62, 110], BigGAN [8], StyleSwin [134], GigaGAN [57], ProGAN [58], ProjectedGAN [109], GANsformer [52], SAN [119], and CIPS [4]), pixel-based diffusion models (e.g., GLIDE [88], ADM [24], and DeepFloyd [22]), latent diffusion models (e.g., VQ-Diffusion [41], Diffusion Transformers [96], and Latent Flow Matching [20]), and an autoregressive model (Taming Transformers [33]).

**Images from commercial models.** We sample 15K images from 11 commercial models using LAION-based captions to evaluate the generalization to state-of-the-art models with typically unknown architectures: DALL·E 2, 3 [91, 92], Ideogram V1, V2 [2], Midjourney V5, V6 [84], Firefly Image 2, 3 [1], FLUX.1-dev, schnell [68], and Imagen 3 [38].

### 3.3. Collecting real images

To help study how real images influence forensics models, we source real images from a variety of existing datasets: LAION [111], ImageNet [23], COCO [74], FFHQ [59], CelebA [76], MetFaces [60], AFHQ [13], Forchheim [44], IMD2020 [89], Landscapes HQ [116], and VISION [115].[3]

### 3.4. Curating the evaluation set

We construct our evaluation set using the incompatible models from our automated sampling pipeline, commercial models (Sec. 3.2), and manually collected open source models. The evaluation set comprises 26K images sampled from 21 models not included in the training set.

[3]Following common convention, we refer to these images as *real* images, even though they may be synthetic (e.g., graphic design). More precisely, our goal is to distinguish "AI-generated" images from the originals.

This includes our commercial models set and an additional 11K images from 10 models: Deci Diffusion V2 [124], GALIP [123], KandinskyV2.2 [114], Kvikontent [66], LCM-LoRA-SDv1.5, LCM-LoRA-SDXL, LCM-LoRA-SSD1B [80], Stable Cascade [97], DF-GAN [122], and HDiT [16], sampled using RAISE [19], ImageNet [23], FFHQ [59], and COCO [74]-based captions.

The generated images are paired with the source real data that are used to prompt the generators. However, since some of the real datasets do not have appropriate licenses for redistribution (e.g., LAION [111, 112]), we created a *public* version of our evaluation set by pairing the generated images with openly licensed COCO [74] and FFHQ [59] which allow redistribution for non-commercial purposes. The *public* version of our evaluation set will serve as an easily reproducible and shareable evaluation set that will complement our default set. We will refer to our default set as the *comprehensive* evaluation set. We also release the instructions to reconstruct our *comprehensive* set. However, note that it may not be possible to exactly reconstruct this set in the future due to link rot.

### 3.5. Generating images

Unconditional models are sampled until we reach the desired number of images. For class conditional models, we sample an equal number of images per class. Text-conditional models are sampled using captions obtained from real images (Sec. 3.3). We either use captions that are already present in the dataset or use BLIP [69] to generate them. Some models such as GigaGAN [57] and HDiT [16] do not provide a pretrained model, so we instead use their pre-generated images. Generated images are saved in PNG format to avoid compression artifacts. However, Firefly [1] generated images are saved in JPEG format as their web UI only allows downloading in JPEG.

## 4. Experiments

We use our dataset to conduct a study of generalization in visual forensics, asking a number of questions: **(1)** How well do forensics models trained on our dataset generalize to unseen models? **(2)** Does adding more models improve detection performance? **(3)** How does diversity of the training data affect performance? **(4)** What architectures and data augmentation schemes are most successful?

### 4.1. Training image forensics models

We train binary classifiers that detect generated images using our dataset to study the generalization in image forensics. We construct our training set of 5.4M images by pairing 2.7M generated images with 2.7M real images.

**Training and evaluation setup.** We evaluate the models trained on our dataset and compare them with prior works [43, 90, 129, 137]. As RED140 [43] already contains RED116 [5], we do not compare with RED116 dataset. Following prior works [90, 129], we use the threshold-independent mean average precision (mAP) and accuracy (Acc.) as our evaluation metrics. We compute the mAP and accuracy by averaging the results of each generative model. We use five evaluation sets: Wang *et al*. [129], Ojha *et al*. [90], Synthbuster [7], GenImage [137], and our evaluation set. All evaluation sets apart from GenImage [137] evaluate out-of-distribution performance for all classifiers. GenImage [137] evaluation set, however, contains the same set of generators used in training, and is an in-distribution evaluation set for their classifiers. Concretely, the evaluation set by Wang *et al*. [129] and Ojha *et al*. [90, 129] contains models such as CRN [10], CycleGAN [136], DALL·E [91], DeepFake [25], IMLE [70], SAN [18], StarGAN [12], and SITD [9] which are unseen by both their and our classifiers. Synthbuster [7] evaluation set is comprised of RAISE [19]-based synthetic images of DALL·E [91, 92], Firefly [1], Glide [88], Midjourney [84], and Stable Diffusion [98, 104], and is mostly out of distribution for all classifiers. GenImage [137] evaluation set is a validation split of their training set; they use an identical set of models to train their classifier: ADM [24], BigGAN [8], Glide [88], Midjourney [84], Stable Diffusion [104], VQ-Diffusion [41], and Wukong [131].

**Model architecture.** Building on prior works which mainly used CLIP-ViT [27, 54, 100] and ResNet-50 [45], we consider ViT [27] and ConvNeXt [77] pretrained models for our classifiers. We use a plain ViT-S backbone [27] pretrained on CLIP objective [54, 100] using LAION-2B [112], ImageNet 21K, and ImageNet 1K datasets [23]. We also experiment with a ConvNeXt-S model [77] pretrained on ImageNet 21K and ImageNet 1K datasets [23]. We replace the classification head with a linear layer with sigmoid activation that outputs the probability of the image being generated. Unlike prior works [15, 90] that freeze the CLIP-ViT backbone, we train the backbone end-to-end. The models are obtained through `timm` [11, 130] library on Hugging Face. We experiment with two input resolutions, $224^2$ and $384^2$, to evaluate the impact of the input resolution on the detector's performance. We denote the detector with $384^2$ input resolution as *High res.* We implement the models using PyTorch [95]. The hyperparameters are detailed in Appendix C.

**Data augmentation.** Prior work considered augmentations that were designed to simulate postprocessing, such as flipping, cropping, Gaussian blur, and JPEG recompression to train their detectors [7, 15, 90, 129]. We propose an augmentation scheme that extends this approach and compare it with previously proposed augmentation methods. We expand the set of augmentations to handle additional transformations that can occur in the wild, such as padding, resizing, rotation, and shear, and integrate them into a framework that can apply complex sequences of transformations. We introduce a modified version of RandAugment [17] that applies a

| Model | Evaluation Set (mAP) | | | | | | | Evaluation Set (Acc) | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Wang *et al*. [129] | Ojha *et al*. [90] | SB [7] | GenImage [137] | Ours Comp. | Ours Public | Mean | Wang *et al*. [129] | Ojha *et al*. [90] | SB [7] | GenImage [137] | Ours Comp. | Ours Public | Mean |
| Wang *et al*. [129] | 0.897 | 0.696 | 0.516 | 0.642 | 0.537 | 0.600 | 0.648 | 0.714 | 0.527 | 0.508 | 0.533 | 0.513 | 0.517 | 0.552 |
| Ojha *et al*. [90] | 0.939 | 0.957 | 0.620 | 0.797 | 0.592 | 0.656 | 0.760 | 0.791 | 0.821 | 0.532 | 0.641 | 0.540 | 0.548 | 0.646 |
| GenImage [137] | 0.929 | 0.984 | 0.813 | **0.999** | 0.912 | 0.968 | 0.934 | 0.795 | 0.966 | 0.719 | **0.990** | 0.818 | 0.886 | 0.862 |
| ♦RED140 [43] - *High res.* | 0.900 | 0.954 | 0.765 | 0.927 | 0.764 | 0.861 | 0.862 | 0.694 | 0.780 | 0.558 | 0.674 | 0.562 | 0.565 | 0.639 |
| Ours | 0.964 | 0.991 | 0.904 | 0.990 | 0.971 | 0.977 | 0.966 | 0.873 | 0.950 | 0.818 | 0.946 | 0.861 | 0.888 | 0.889 |
| Ours - *High res.* | **0.967** | **0.996** | **0.974** | 0.998 | **0.987** | **0.994** | **0.986** | **0.901** | **0.970** | **0.908** | 0.957 | **0.892** | **0.912** | **0.923** |

Table 2. **Generalization of AI-generated image detectors across benchmarks.** We evaluate the classifiers trained on our dataset on several benchmarks, including our own. We also evaluate several previously released classifiers. Our *Comprehensive* set (abbreviated as *Comp.*) pairs the generated images with original real data; the *Public* set pairs them with openly licensed COCO [74] and FFHQ [59] for license-compliant redistribution of the evaluation set (Sec. 3.4). We use plain CLIP-ViT-S [27, 54, 100] architecture with $224^2$ and $384^2$ (*High res.*) input resolutions, Wang *et al*. [129] and GenImage [137] use ResNet-50 [45] with $224^2$ input resolution, and Ojha *et al*. [90] uses CLIP-ViT-L with $224^2$ input resolution as the backbone. Our classifiers show robust performance across all evaluation sets, outperforming all baselines in out-of-distribution settings ([7, 90, 129] and *Ours*) and nearly matches GenImage [137] on its *in-distribution* evaluation set. ♦: RED140 [43] is trained following our training procedure with $384^2$ input resolution as they do not provide a trained classifier.

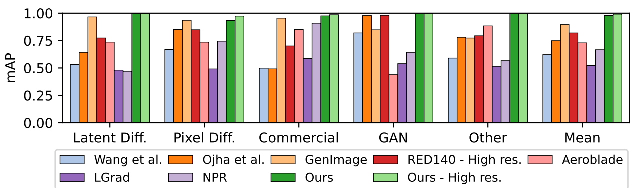



Figure 3. **Performance across generator types.** We evaluate the classifier performance across five generator types – latent diffusion, pixel diffusion, commercial models, GANs, and other architecture (Stable Cascade [97]), and report the mean. We additionally evaluate three forensics methods, Aeroblade [103], LGrad [120], and NPR [121]. Our classifiers show robust performance across all generator types, whereas prior works struggle to generalize.

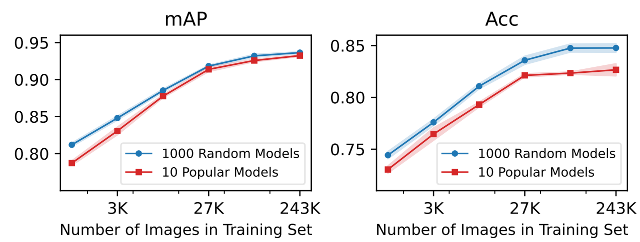



Figure 4. **Performance with increasing number of images.** We train a classifier with varying numbers of images from two sets: 1000 randomly chosen models and 10 popular (highly downloaded) models in the systematically collected subset. We report the mean and standard error bands for each data point across 4 randomly sampled subsets. The classifier trained from 1000 random models outperforms 10 popular models in all cases. Notably, the accuracy gap is wider than that of mAP, which may suggest that having a diverse set of models improves accuracy threshold calibration.

randomly-ordered sequence of augmentations to the images. Specifically, our modified RandAugment samples a random number $n$ between 0 and $n_{\mathtt{max}}$ for each augmentation type. Then, it applies the augmentations in random order until $n$ augmentations are applied for each augmentation type to the image. We use various augmentations, including in-memory JPEG compression, random resizing with random interpolation methods, cropping, flipping, rotation, translation, shear, padding, and cutout.

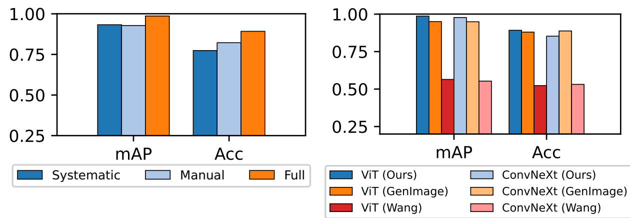
## 4.2. Generalization across benchmarks

We first evaluate how well classifiers trained on our dataset perform across benchmarks. In Table 2, we observe that our models outperform the prior works [43, 90, 129, 137] on all evaluation sets except GenImage. This is expected since the GenImage evaluation set is a validation split of their training set; all of the generators are already seen by their classifier. On all other *unseen* benchmarks, our classifiers outperform all prior works. Notably, our classifiers achieve very high performance (0.987 mAP and 89.2% accuracy) on our out-of-distribution, *comprehensive* evaluation set, with a significant margin over prior works. This gap in performance can be traced to our training data which incorporates a substantially richer variety of generators than existing works. Consequently, our classifiers demonstrate robust generalization to out-of-distribution data, where prior works often struggle.

To better illustrate the generalization of the classifiers, in Figure 3, we split our *comprehensive* evaluation set into five subsets and evaluate them: latent diffusion, pixel diffusion, commercial models, GANs, and other architecture type (Stable Cascade [97]). Furthermore, we evaluate three additional forensics methods: Aeroblade [103], LGrad [120], and NPR [121]. Our classifiers show strong performance across all generator types, unlike prior works which struggle to generalize. For the following experiments, we use our best-performing model (*High res.*) unless stated otherwise.

## 4.3. Impact of model diversity

Next, we examine the impact of the number of models in training data. We train classifiers with images sampled from 3 to 3333 generators and evaluate them (Fig. 1). To ensure that the gains are not due to simply sampling qualita-



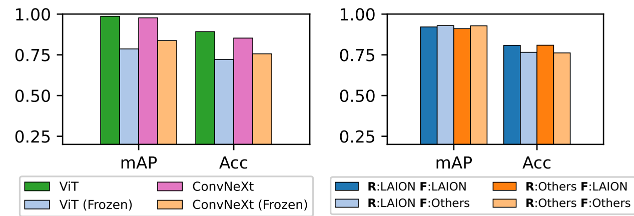

(a) Impact of generator type diversity (b) Classifier backbone comparison

Figure 5. (a) **Performance and model diversity.** We compare detection performance for commercial models using classifiers trained on different subsets of the dataset: the systematically collected latent diffusion models, the manually chosen models containing diverse generator types, and both. As diversity increases, so does performance. (b) **Classifier backbone comparison.** We compare the architectures across datasets: ours, GenImage [137], and Wang *et al*. [129]. Performance is similar between architectures.

tively different architectures, we only use our systematically collected latent diffusion models. For each data point, we sample 10 random subsets of models with 100K training images each and report the mean and standard error bands. We use an extended evaluation set that includes non-latent diffusion generators from our training set, which allows us to comprehensively assess the generalization capability of the classifiers trained exclusively on latent diffusion models. We find that the performance steadily increases with the number of models. However, the performance begins to flatten out beyond 1000 models, suggesting diminishing returns. Interestingly, the performance also improves on out-of-distribution architectures such as GANs and pixel-based diffusion models, even though the classifier is only trained on latent diffusion models.

In Figure 4, we vary the number of images from two sets: 1000 randomly chosen models and 10 popular models (as denoted by their number of downloads) downloaded from our systematically collected diffusion models. While the results show that the performance improves with more training images, it begins to plateau at approximately 27K images. Moreover, the classifier trained on 1000 models outperforms the 10 models in all cases, indicating that model diversity is important for strong performance. We also note that the accuracy gap is noticeably wider than that of mAP, which may suggest that model diversity is crucial in calibrating the accuracy thresholds of the classifiers.

Our experiments show that the performance improvements from increasing the number of models may plateau when they are limited to a single generator type (Fig. 1). In Figure 5a, we show that the diversity of the generator type also plays a major role in generalization. We train classifiers on three different sets of training data: our systematically collected set, manually chosen set, and a full set consisting of both subsets. The *systematic* set comprises entirely of latent diffusion models, and the *manual* set contains numerous generator types, including GANs, latent and pixel-based diffusion, and autoregressive models (Sec. 3.2). The classifier trained on the *manual* set with more diverse generator types shows similar performance compared to the one trained on the *systematic* set, despite containing fewer than half the number of images (774K *vs*. 1.9M). Additionally, we find that the two sets are complementary; the performance is further improved when we train using both sets.




(a) Impact of frozen backbones (b) Semantic alignment analysis

Figure 6. (a) **Evaluating frozen backbones.** Freezing the pretrained backbone, a common practice in prior works [15, 90], consistently decreases the performance. (b) **Analyzing source and generated data alignment.** We evaluate how the pairing of the real datasets affects performance. '**R**' denotes the real dataset used in training, and '**F**' indicates the source dataset used to obtain the captions for prompting the generators. The results suggest that pairing the source data (i.e., real data used to prompt the generators) with the generated images is not essential for performance.

### 4.4. Analysis of design choices

We examine the impact of various design choices, including some suggested in earlier works. In particular, we investigate the choice of backbone models, freezing the backbone, semantic alignment between the real and generated data, and robustness to transformations.

**Classifier backbone comparison.** We compare the performance of the classifier trained using CLIP-ViT [27, 54, 100] and ConvNeXt [77] backbones following our training procedure in Figure 5b. We examine three datasets: ours, GenImage [137], and Wang *et al*. [129]. We observe similar performance between architectures across all datasets.

**Frozen backbone.** Prior works [15, 90] suggested using a frozen CLIP-ViT backbone for training the classifiers. We study this by training the classifiers with both frozen and unfrozen pretrained backbones, using CLIP-ViT [27, 54, 100] and ConvNeXt [77]. As shown in Fig. 6a, freezing the backbone consistently leads to worse results, indicating that end-to-end training is crucial for high performance.

**Semantic alignment.** Existing works often pair the generated images with the source dataset (i.e., the real dataset used to prompt or generate the images) arguing that misaligned data can introduce bias [7, 15, 90, 129]. We test this practice in Fig. 6b by examining the performance with both semantically aligned and misaligned real datasets. Specifically, we consider two real datasets: one comprised exclusively from LAION [111] and another combining ImageNet [23],

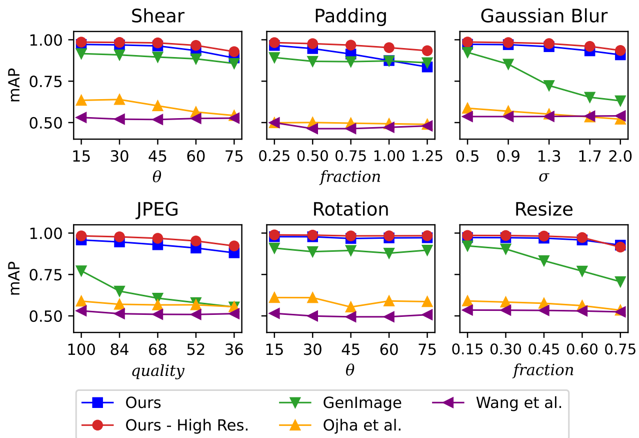



Figure 7. **Robustness to various transformations.** Our classifiers display robust performance across transformations. Other works generally show more sensitivity to these factors.

MS-COCO [74], LandscapesHQ [116], Forchheim [44], VISION [115], and IMD2020 [89]. We sample our systematically collected latent diffusion models using these two sets and categorize the generated images by their source dataset. The performance differences of all pairs were marginal, suggesting that strict alignment may not be as critical with sufficiently large data.

**Robustness to transformations.** Figure 7 illustrates the robustness of the classifiers against various transformations. Following prior works [90, 129], we test robustness to JPEG compression and Gaussian blur. Additionally, we examine robustness to rotation, resizing, padding, and shear, as they commonly occur in real-world scenarios. For *padding*, we randomly pad the width or height of the image with a given fraction and scale it back to the original size. Similarly, in *resize*, we randomly upsample or downsample the height and width of the image by a given fraction and resize it back to the original size (e.g., if *fraction* is $0.3$, resize the height and width to $0.7\times$ or $1.3\times$ and then scale it back to the original size). The results demonstrate that our models are more robust to transformations than existing models. Specifically, GenImage [137] is notably more sensitive to Gaussian blur, JPEG compression, and resizing artifacts; classifier by Ojha *et al*. [90] displays sensitivity to *shear* transforms, and the one by Wang *et al*. [129] performs poorly overall.

### 4.5. Other forensics applications

Our dataset may enable further forensics studies that can take advantage of our diverse array of generators. To illustrate this, in Appendix A, we identify the type of generator used to synthesize a given image by using $k$-nearest neighbor in the feature space of our classifier.

## 5. Conclusion

In this paper, we studied the problem of generalizing to unseen generative models in synthetic image detection. We proposed a new dataset, *Community Forensics*, which contains 4803 models and 2.7M images collected from various public sources. We studied the impact of model diversity and demonstrated that it plays a crucial role in enhancing data diversity and generalization performance. We trained classifiers on our dataset, studied their ability to generalize in various settings, and evaluated previously proposed models and training practices.

We do not intend for our dataset to be used to train classifiers that are directly used in the wild. Detecting in-the-wild synthetic images remains a challenging open problem, and detection errors can have severe consequences (e.g., falsely accusing an author of creating fake images or allowing misinformation to be certified as real). We hope that our work will serve as a stepping stone for future research in this area by providing tools and insights for studying generalization and data collection strategies.

**Limitations.** While our dataset is diverse, a large portion of the data is diffusion-based, especially models based on Stable Diffusion [104]. Despite this, Figure 1 demonstrates that adding more diffusion models to the training data improves the generalization capability of the classifiers across other generator types. Furthermore, Figure 5a illustrates improved classifier performance when combining the diffusion-only *systematic* set with a more diverse *manual* set. These results suggest that despite their similarities, each diffusion model may uniquely contribute through its semantic content, image processing pipelines, or architectural variations (Sec. 3.1). Future work may explore collecting more diverse models, including GANs, VQ-VAEs [126], and autoregressive models. We also note that the generative models sourced from the community may contain inappropriate content. While in many contexts it is important to detect such images, these generated images may require further scrutiny before being used in other downstream applications. Finally, although our experiments suggest that our forensics classifiers generalize to unseen models better than those of previous work, their error rates are still too high for many critical applications.

**Acknowledgements.** We thank the creators of the many open source models that we used to collect the Community Forensics dataset. We thank Chenhao Zheng, Cameron Johnson, Matthias Kirchner, Daniel Geng, Ziyang Chen, Ayush Shrivastava, Yiming Dou, Chao Feng, Xuanchen Lu, Zihao Wei, Zixuan Pan, Inbum Park, Rohit Banerjee, and Ang Cao for the valuable discussions and feedback. This research was developed with funding from the Defense Advanced Research Projects Agency (DARPA) under Contract No. HR001120C0123.

## References


[1] Adobe. Firefly. https://www.adobe.com/products/firefly, 2023. 4, 5

[2] Ideogram AI. Ideogram. https://ideogram.ai, 2024. 4

[3] Rushil Anirudh and Jayaraman J Thiagarajan. Out of distribution detection via neural network anchoring. In *Asian Conference on Machine Learning*, pages 32–47. PMLR, 2023. 3

[4] Ivan Anokhin, Kirill Demochkin, Taras Khakhulin, Gleb Sterkin, Victor Lempitsky, and Denis Korzhenkov. Image generators with conditionally-independent pixel synthesis. *arXiv preprint arXiv:2011.13775*, 2020. 4

[5] Vishal Asnani, Xi Yin, Tal Hassner, and Xiaoming Liu. Reverse engineering of generative models: Inferring model hyperparameters from generated images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(12):15477–15493, 2023. 2, 3, 4, 5

[6] Anton Bakhtin, Sam Gross, Myle Ott, Yuntian Deng, Marc'Aurelio Ranzato, and Arthur Szlam. Real or fake? learning to discriminate machine from human generated text. *arXiv preprint arXiv:1906.03351*, 2019. 3

[7] Quentin Bammey. Synthbuster: Towards detection of diffusion model generated images. *IEEE Open Journal of Signal Processing*, 2023. 2, 3, 4, 5, 6, 7

[8] Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale GAN training for high fidelity natural image synthesis. In *International Conference on Learning Representations*, 2019. 3, 4, 5

[9] Chen Chen, Qifeng Chen, Jia Xu, and Vladlen Koltun. Learning to see in the dark. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3291–3300, 2018. 3, 5

[10] Qifeng Chen and Vladlen Koltun. Photographic image synthesis with cascaded refinement networks. In *Proceedings of the IEEE international conference on computer vision*, pages 1511–1520, 2017. 3, 5

[11] Mehdi Cherti, Romain Beaumont, Ross Wightman, Mitchell Wortsman, Gabriel Ilharco, Cade Gordon, Christoph Schuhmann, Ludwig Schmidt, and Jenia Jitsev. Reproducible scaling laws for contrastive language-image learning. *arXiv preprint arXiv:2212.07143*, 2022. 5

[12] Yunjey Choi, Minje Choi, Munyoung Kim, Jung-Woo Ha, Sunghun Kim, and Jaegul Choo. Stargan: Unified generative adversarial networks for multi-domain image-to-image translation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018. 3, 5

[13] Yunjey Choi, Youngjung Uh, Jaejun Yoo, and Jung-Woo Ha. Stargan v2: Diverse image synthesis for multiple domains. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2020. 4

[14] Riccardo Corvi, Davide Cozzolino, Giada Zingarini, Giovanni Poggi, Koki Nagano, and Luisa Verdoliva. On the detection of synthetic images generated by diffusion models. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE, 2023. 3

[15] Davide Cozzolino, Giovanni Poggi, Riccardo Corvi, Matthias Nießner, and Luisa Verdoliva. Raising the bar of ai-generated image detection with clip. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4356–4366, 2024. 2, 3, 4, 5, 7

[16] Katherine Crowson, Stefan Andreas Baumann, Alex Birch, Tanishq Mathew Abraham, Daniel Z Kaplan, and Enrico Shippole. Scalable high-resolution pixel-space image synthesis with hourglass diffusion transformers. In *Proceedings of the 41st International Conference on Machine Learning*, pages 9550–9575. PMLR, 2024. 5, 15

[17] Ekin Dogus Cubuk, Barret Zoph, Jon Shlens, and Quoc Le. Randaugment: Practical automated data augmentation with a reduced search space. In *Advances in Neural Information Processing Systems*, pages 18613–18624. Curran Associates, Inc., 2020. 5

[18] Tao Dai, Jianrui Cai, Yongbing Zhang, Shu-Tao Xia, and Lei Zhang. Second-order attention network for single image super-resolution. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11065–11074, 2019. 3, 5

[19] Duc-Tien Dang-Nguyen, Cecilia Pasquini, Valentina Conotter, and Giulia Boato. Raise: A raw images dataset for digital image forensics. In *Proceedings of the 6th ACM multimedia systems conference*, pages 219–224, 2015. 5, 15

[20] Quan Dao, Hao Phung, Binh Nguyen, and Anh Tran. Flow matching in latent space. *arXiv preprint arXiv:2307.08698*, 2023. 4

[21] Tiago José De Carvalho, Christian Riess, Elli Angelopoulou, Helio Pedrini, and Anderson de Rezende Rocha. Exposing digital image forgeries by illumination color classification. *IEEE Transactions on Information Forensics and Security*, 8(7):1182–1194, 2013. 3

[22] DeepFloyd. Deepfloyd. https://huggingface.co/DeepFloyd/IF-I-L-v1.0, 2024. 4

[23] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. 4, 5, 7, 15

[24] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34:8780–8794, 2021. 4, 5

[25] Brian Dolhansky, Joanna Bitton, Ben Pflaum, Jikuo Lu, Russ Howes, Menglin Wang, and Cristian Canton Ferrer. The deepfake detection challenge (dfdc) dataset. *arXiv preprint arXiv:2006.07397*, 2020. 2, 5

[26] Jing Dong, Wei Wang, and Tieniu Tan. Casia image tampering detection evaluation database. In *2013 IEEE China summit and international conference on signal and information processing*, pages 422–426. IEEE, 2013. 3

[27] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. *ICLR*, 2021. 5, 6, 7

[28] Yilun Du, Shuang Li, Joshua Tenenbaum, and Igor Mordatch. Improved contrastive divergence training of energy-based models. In *International Conference on Machine Learning*, pages 2837–2848. PMLR, 2021. 3

[29] Ricard Durall, Margret Keuper, and Janis Keuper. Watch your up-convolution: Cnn based generative deep neural networks are failing to reproduce spectral distributions. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 7890–7899, 2020. 3

[30] Tarik Dzanic, Karan Shah, and Freddie Witherden. Fourier spectrum discrepancies in deep network generated images. *Advances in neural information processing systems*, 33:3022–3032, 2020. 3

[31] Sven Elflein, Bertrand Charpentier, Daniel Zügner, and Stephan Günnemann. On out-of-distribution detection with energy-based models, 2021. 3

[32] David C Epstein, Ishan Jain, Oliver Wang, and Richard Zhang. Online detection of ai-generated images. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 382–392, 2023. 2, 3, 4

[33] Patrick Esser, Robin Rombach, and Bjorn Ommer. Taming transformers for high-resolution image synthesis. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12873–12883, 2021. 4

[34] Hugging Face. Hugging face diffusers library. https://huggingface.co/models?library=diffusers, accessed on June 05, 2022, 2022. 2, 3, 4, 16

[35] Sebastian Gehrmann, Hendrik Strobelt, and Alexander M Rush. Gltr: Statistical detection and visualization of generated text. *arXiv preprint arXiv:1906.04043*, 2019. 3

[36] Thomas Gloe and Rainer Böhme. The'dresden image database'for benchmarking digital image forensics. In *Proceedings of the 2010 ACM symposium on applied computing*, pages 1584–1590, 2010. 4

[37] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *Advances in Neural Information Processing Systems*, 27, 2014. 2

[38] Google. Imagen 3. https://deepmind.google/technologies/imagen-3, 2024. 4

[39] Karol Gregor, Ivo Danihelka, Andriy Mnih, Charles Blundell, and Daan Wierstra. Deep autoregressive networks. In *International Conference on Machine Learning*, pages 1242–1250. PMLR, 2014. 2

[40] Patrick Grommelt, Louis Weiss, Franz-Josef Pfreundt, and Janis Keuper. Fake or jpeg? revealing common biases in generated image detection datasets. *arXiv preprint arXiv:2403.17608*, 2024. 3

[41] Shuyang Gu, Dong Chen, Jianmin Bao, Fang Wen, Bo Zhang, Dongdong Chen, Lu Yuan, and Baining Guo. Vector quantized diffusion model for text-to-image synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10696–10706, 2022. 4, 5

[42] Fabrizio Guillaro, Giada Zingarini, Ben Usman, Avneesh Sud, Davide Cozzolino, and Luisa Verdoliva. A bias-free training paradigm for more general ai-generated image detection, 2024. 3

[43] Xiao Guo, Vishal Asnani, Sijia Liu, and Xiaoming Liu. Tracing hyperparameter dependencies for model parsing via learnable graph pooling network. *Advances in Neural Information Processing Systems*, 37:116899–116932, 2024. 2, 3, 4, 5, 6

[44] Benjamin Hadwiger and Christian Riess. The forchheim image database for camera identification in the wild. In *Pattern Recognition. ICPR International Workshops and Challenges: Virtual Event, January 10–15, 2021, Proceedings, Part VI*, pages 500–515. Springer, 2021. 4, 8

[45] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 5, 6

[46] Dan Hendrycks and Kevin Gimpel. A baseline for detecting misclassified and out-of-distribution examples in neural networks. In *International Conference on Learning Representations*, 2016. 3

[47] Dan Hendrycks, Mantas Mazeika, Saurav Kadavath, and Dawn Song. Using self-supervised learning can improve model robustness and uncertainty. In *Advances in Neural Information Processing Systems*. Curran Associates, Inc., 2019. 3

[48] Dan Hendrycks, Steven Basart, Norman Mu, Saurav Kadavath, Frank Wang, Evan Dorundo, Rahul Desai, Tyler Zhu, Samyak Parajuli, Mike Guo, Dawn Song, Jacob Steinhardt, and Justin Gilmer. The many faces of robustness: A critical analysis of out-of-distribution generalization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 8340–8349, 2021. 2, 3

[49] Yan Hong and Jianfu Zhang. Wildfake: A large-scale challenging dataset for ai-generated images detection. *arXiv preprint arXiv:2402.11843*, 2024. 3

[50] Shuwei Hou, Yan Ju, Chengzhe Sun, Shan Jia, Lipeng Ke, Riky Zhou, Anita Nikolich, and Siwei Lyu. Deepfake-o-meter v2. 0: An open platform for deepfake detection. *arXiv preprint arXiv:2404.13146*, 2024. 3

[51] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021. 4

[52] Drew A Hudson and Larry Zitnick. Generative adversarial transformers. In *International conference on machine learning*, pages 4487–4499. PMLR, 2021. 4

[53] Minyoung Huh, Andrew Liu, Andrew Owens, and Alexei A Efros. Fighting fake news: Image splice detection via learned self-consistency. In *Proceedings of the European conference on computer vision (ECCV)*, 2018. 3

[54] Gabriel Ilharco, Mitchell Wortsman, Ross Wightman, Cade Gordon, Nicholas Carlini, Rohan Taori, Achal Dave, Vaishaal Shankar, Hongseok Namkoong, John Miller, Hannaneh Hajishirzi, Ali Farhadi, and Ludwig Schmidt. Openclip, 2021. If you use this software, please cite it as below. 5, 6, 7

[55] Ganesh Jawahar, Muhammad Abdul-Mageed, and Laks VS Lakshmanan. Automatic detection of machine generated text: A critical survey. *arXiv preprint arXiv:2011.01314*, 2020. 3

[56] Micah K Johnson and Hany Farid. Exposing digital forgeries in complex lighting environments. *IEEE Transactions on Information Forensics and Security*, 2(3):450–461, 2007. 3

[57] Minguk Kang, Jun-Yan Zhu, Richard Zhang, Jaesik Park, Eli Shechtman, Sylvain Paris, and Taesung Park. Scaling up gans for text-to-image synthesis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 4, 5

[58] 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*, 2018. 3, 4

[59] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4401–4410, 2019. 3, 4, 5, 6, 15

[60] Tero Karras, Miika Aittala, Janne Hellsten, Samuli Laine, Jaakko Lehtinen, and Timo Aila. Training generative adversarial networks with limited data. *Advances in neural information processing systems*, 33:12104–12114, 2020. 4

[61] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of stylegan. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8110–8119, 2020.

[62] Tero Karras, Miika Aittala, Samuli Laine, Erik Härkönen, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Alias-free generative adversarial networks. *Advances in neural information processing systems*, 34:852–863, 2021. 4

[63] Hasam Khalid, Shahroz Tariq, Minha Kim, and Simon Woo. Fakeavceleb: A novel audio-video multimodal deepfake dataset. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*, 2021. 2

[64] P. Korus and J. Huang. Multi-scale analysis strategies in prnu-based tampering localization. *IEEE Trans. on Information Forensics & Security*, 2017. 3

[65] Kalpesh Krishna, Yixiao Song, Marzena Karpinska, John Wieting, and Mohit Iyyer. Paraphrasing evades detectors of ai-generated text, but retrieval is an effective defense. *Advances in Neural Information Processing Systems*, 36, 2024. 3

[66] Kvikontent. Kvikontent-midjourney v6. https://huggingface.co/Kvikontent/midjourney-v6, 2023. 5

[67] Patrick Kwon, Jaeseong You, Gyuhyeon Nam, Sungwoo Park, and Gyeongsu Chae. Kodf: A large-scale korean deepfake detection dataset. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021. 2

[68] Black Forst Labs. Flux. `https://blackforestlabs.ai`, 2024. 4

[69] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *ICML*, 2022. 5

[70] Ke Li, Tianhao Zhang, and Jitendra Malik. Diverse image synthesis from semantic layouts via conditional imle. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4220–4229, 2019. 3, 5

[71] Yuezun Li, Xin Yang, Pu Sun, Honggang Qi, and Siwei Lyu. Celeb-df: A large-scale challenging dataset for deepfake forensics. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020. 2

[72] Yewen Li, Chaojie Wang, Xiaobo Xia, Tongliang Liu, Bo An, et al. Out-of-distribution detection with an adaptive likelihood ratio on informative hierarchical vae. *Advances in Neural Information Processing Systems*, 35:7383–7396, 2022. 3

[73] Shiyu Liang, Yixuan Li, and R Srikant. Enhancing the reliability of out-of-distribution image detection in neural networks. In *International Conference on Learning Representations*, 2018. 3

[74] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, pages 740–755. Springer, 2014. 4, 5, 6, 8, 15

[75] Weitang Liu, Xiaoyun Wang, John Owens, and Yixuan Li. Energy-based out-of-distribution detection. In *Advances in Neural Information Processing Systems*, pages 21464–21475. Curran Associates, Inc., 2020. 3

[76] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proceedings of International Conference on Computer Vision (ICCV)*, 2015. 4, 15

[77] Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. A convnet for the 2020s. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11976–11986, 2022. 5, 7

[78] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019. 15

[79] Simian Luo, Yiqin Tan, Longbo Huang, Jian Li, and Hang Zhao. Latent consistency models: Synthesizing high-resolution images with few-step inference. *arXiv preprint arXiv:2310.04378*, 2023. 4

[80] Simian Luo, Yiqin Tan, Suraj Patil, Daniel Gu, Patrick von Platen, Apolinário Passos, Longbo Huang, Jian Li, and Hang Zhao. Lcm-lora: A universal stable-diffusion acceleration module. *arXiv preprint arXiv:2311.05556*, 2023. 2, 5

[81] Francesco Marra, Diego Gragnaniello, Luisa Verdoliva, and Giovanni Poggi. Do gans leave artificial fingerprints? In *2019 IEEE conference on multimedia information processing and retrieval (MIPR)*, pages 506–511. IEEE, 2019. 3

[82] Leland McInnes, John Healy, Nathaniel Saul, and Lukas Großberger. Umap: Uniform manifold approximation and projection. *Journal of Open Source Software*, 3(29), 2018. 15

[83] Paulius Micikevicius, Sharan Narang, Jonah Alben, Gregory Diamos, Erich Elsen, David Garcia, Boris Ginsburg, Michael Houston, Oleksii Kuchaiev, Ganesh Venkatesh, et al. Mixed precision training. In *International Conference on Learning Representations*, 2018. 15

[84] Inc. Midjourney. Midjourney. `https://www.midjourney.com/home`, 2022. 4, 5

[85] Eric Mitchell, Yoonho Lee, Alexander Khazatsky, Christopher D Manning, and Chelsea Finn. Detectgpt: Zero-shot machine-generated text detection using probability curvature. In *International Conference on Machine Learning*, pages 24950–24962. PMLR, 2023. 3

[86] Sina Mohseni, Mandar Pitale, JBS Yadawa, and Zhangyang Wang. Self-supervised learning for generalizable out-of-distribution detection. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(04):5216–5223, 2020. 3

[87] Tian-Tsong Ng, Shih-Fu Chang, and Q Sun. A data set of authentic and spliced image blocks. *Columbia University, ADVENT Technical Report*, 4, 2004. 3

[88] Alexander Quinn Nichol, Prafulla Dhariwal, Aditya Ramesh, Pranav Shyam, Pamela Mishkin, Bob Mcgrew, Ilya Sutskever, and Mark Chen. Glide: Towards photorealistic image generation and editing with text-guided diffusion models. In *International Conference on Machine Learning*, pages 16784–16804. PMLR, 2022. 4, 5

[89] Adam Novozamsky, Babak Mahdian, and Stanislav Saic. Imd2020: A large-scale annotated dataset tailored for detecting manipulated images. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision Workshops*, pages 71–80, 2020. 4, 8

[90] Utkarsh Ojha, Yuheng Li, and Yong Jae Lee. Towards universal fake image detectors that generalize across generative models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 24480–24489, 2023. 1, 2, 3, 4, 5, 6, 7, 8, 15

[91] OpenAI. Dall-e 2. https://openai.com/index/dall-e-2, 2022. 4, 5

[92] OpenAI. Dall-e 3. https://openai.com/index/dall-e-3, 2023. 4, 5

[93] CreativeML OpenRail-M License. Creativeml openrail-m license. https://huggingface.co/spaces/CompVis/stable-diffusion-license. 15, 16

[94] Taesung Park, Ming-Yu Liu, Ting-Chun Wang, and Jun-Yan Zhu. Semantic image synthesis with spatially-adaptive normalization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019. 3

[95] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019. 5

[96] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4195–4205, 2023. 4

[97] Pablo Pernias, Dominic Rampas, Mats Leon Richter, Christopher Pal, and Marc Aubreville. Würstchen: An efficient architecture for large-scale text-to-image diffusion models. In *The Twelfth International Conference on Learning Representations*, 2023. 5, 6, 15

[98] Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. Sdxl: Improving latent diffusion models for high-resolution image synthesis. In *The Twelfth International Conference on Learning Representations*, 2023. 5

[99] Alin C Popescu and Hany Farid. Exposing digital forgeries by detecting traces of resampling. *IEEE Transactions on signal processing*, 2005. 3

[100] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021. 3, 5, 6, 7

[101] Anirudh Sundara Rajan, Utkarsh Ojha, Jedidiah Schloesser, and Yong Jae Lee. Aligned datasets improve detection of latent diffusion-generated images, 2025. 3

[102] Jie Ren, Peter J Liu, Emily Fertig, Jasper Snoek, Ryan Poplin, Mark Depristo, Joshua Dillon, and Balaji Lakshminarayanan. Likelihood ratios for out-of-distribution detection. *Advances in neural information processing systems*, 32, 2019. 3

[103] Jonas Ricker, Denis Lukovnikov, and Asja Fischer. Aeroblade: Training-free detection of latent diffusion images using autoencoder reconstruction error. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9130–9140, 2024. 6

[104] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10684–10695, 2022. 2, 4, 5, 8

[105] Andreas Rössler, Davide Cozzolino, Luisa Verdoliva, Christian Riess, Justus Thies, and Matthias Nießner. Faceforensics: A large-scale video dataset for forgery detection in human faces. *arXiv preprint arXiv:1803.09179*, 2018. 1, 2

[106] Andreas Rossler, Davide Cozzolino, Luisa Verdoliva, Christian Riess, Justus Thies, and Matthias Nießner. Faceforensics++: Learning to detect manipulated facial images. In *Proceedings of the IEEE/CVF international conference on computer vision*, 2019. 1, 2, 3

[107] Vinu Sankar Sadasivan, Aounon Kumar, Sriram Balasubramanian, Wenxiao Wang, and Soheil Feizi. Can ai-generated text be reliably detected? *arXiv preprint arXiv:2303.11156*, 2023. 3

[108] Chandramouli Shama Sastry and Sageev Oore. Detecting out-of-distribution examples with Gram matrices. In *Proceedings of the 37th International Conference on Machine Learning*, pages 8491–8501. PMLR, 2020. 3

[109] Axel Sauer, Kashyap Chitta, Jens Müller, and Andreas Geiger. Projected gans converge faster. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021. 4

[110] Axel Sauer, Katja Schwarz, and Andreas Geiger. Stylegan-xl: Scaling stylegan to large diverse datasets. In *ACM SIGGRAPH 2022 conference proceedings*, pages 1–10, 2022. 4

[111] Christoph Schuhmann, Richard Vencu, Romain Beaumont, Robert Kaczmarczyk, Clayton Mullis, Aarush Katta, Theo Coombes, Jenia Jitsev, and Aran Komatsuzaki. Laion-400m:

Open dataset of clip-filtered 400 million image-text pairs. *arXiv preprint arXiv:2111.02114*, 2021. 2, 4, 5, 7, 15

[112] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade W Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, Patrick Schramowski, Srivatsa R Kundurthy, Katherine Crowson, Ludwig Schmidt, Robert Kaczmarczyk, and Jenia Jitsev. LAION-5b: An open large-scale dataset for training next generation image-text models. In *Thirty-sixth Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2022. 2, 3, 4, 5

[113] Vikash Sehwag, Mung Chiang, and Prateek Mittal. Ssd: A unified framework for self-supervised outlier detection. In *International Conference on Learning Representations*, 2020. 3

[114] Arseniy Shakhmatov, Anton Razzhigaev, Aleksandr Nikolich, Vladimir Arkhipkin, Igor Pavlov, Andrey Kuznetsov, and Denis Dimitrov. Kandinsky 2.2. https://github.com/ai-forever/Kandinsky-2, 2023. 5

[115] Dasara Shullani, Marco Fontani, Massimo Iuliani, Omar Al Shaya, and Alessandro Piva. Vision: a video and image dataset for source identification. *EURASIP Journal on Information Security*, 2017:1–16, 2017. 4, 8

[116] Ivan Skorokhodov, Grigorii Sotnikov, and Mohamed Elhoseiny. Aligning latent and image spaces to connect the unconnectable. *arXiv preprint arXiv:2104.06954*, 2021. 4, 8, 15

[117] Irene Solaiman, Miles Brundage, Jack Clark, Amanda Askell, Ariel Herbert-Voss, Jeff Wu, Alec Radford, Gretchen Krueger, Jong Wook Kim, Sarah Kreps, et al. Release strategies and the social impacts of language models. *arXiv preprint arXiv:1908.09203*, 2019. 3

[118] Yang Song, Prafulla Dhariwal, Mark Chen, and Ilya Sutskever. Consistency models. In *Proceedings of the 40th International Conference on Machine Learning*, pages 32211–32252, 2023. 2

[119] Yuhta Takida, Masaaki Imaizumi, Takashi Shibuya, Chieh-Hsin Lai, Toshimitsu Uesaka, Naoki Murata, and Yuki Mitsufuji. SAN: Inducing metrizability of GAN with discriminative normalized linear layer. In *The Twelfth International Conference on Learning Representations*, 2024. 4

[120] Chuangchuang Tan, Yao Zhao, Shikui Wei, Guanghua Gu, and Yunchao Wei. Learning on gradients: Generalized artifacts representation for gan-generated images detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12105–12114, 2023. 6

[121] Chuangchuang Tan, Yao Zhao, Shikui Wei, Guanghua Gu, Ping Liu, and Yunchao Wei. Rethinking the up-sampling operations in cnn-based generative network for generalizable deepfake detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 28130–28139, 2024. 6

[122] Ming Tao, Hao Tang, Fei Wu, Xiao-Yuan Jing, Bing-Kun Bao, and Changsheng Xu. Df-gan: A simple and effective baseline for text-to-image synthesis. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16515–16525, 2022. 5

[123] Ming Tao, Bing-Kun Bao, Hao Tang, and Changsheng Xu. Galip: Generative adversarial clips for text-to-image synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14214–14223, 2023. 5

[124] DeciAI Research Team. Decidiffusion 2.0, 2024. 5

[125] Adaku Uchendu, Thai Le, Kai Shu, and Dongwon Lee. Authorship attribution for neural text generation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8384–8395, 2020. 3

[126] Aaron Van Den Oord, Oriol Vinyals, et al. Neural discrete representation learning. *Advances in neural information processing systems*, 30, 2017. 8

[127] Patrick von Platen, Suraj Patil, Anton Lozhkov, Pedro Cuenca, Nathan Lambert, Kashif Rasul, Mishig Davaadorj, Dhruv Nair, Sayak Paul, William Berman, Yiyi Xu, Steven Liu, and Thomas Wolf. Diffusers: State-of-the-art diffusion models. https://github.com/huggingface/diffusers, 2022. 2, 3, 4

[128] Apoorv Vyas, Nataraj Jammalamadaka, Xia Zhu, Dipankar Das, Bharat Kaul, and Theodore L. Willke. Out-of-distribution detection using an ensemble of self supervised leave-out classifiers. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018. 3

[129] Sheng-Yu Wang, Oliver Wang, Richard Zhang, Andrew Owens, and Alexei A. Efros. Cnn-generated images are surprisingly easy to spot... for now. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 1, 2, 3, 4, 5, 6, 7, 8

[130] Ross Wightman. Pytorch image models. https://github.com/huggingface/pytorch-image-models, 2019. 5

[131] Wukong. https://xihe.mindspore.cn/modelzoo/wukong, 2022. 5

[132] Zhisheng Xiao, Qing Yan, and Yali Amit. Likelihood regret: An out-of-distribution detection score for variational autoencoder. In *Advances in Neural Information Processing Systems*, pages 20685–20696. Curran Associates, Inc., 2020. 3

[133] Fisher Yu, Yinda Zhang, Shuran Song, Ari Seff, and Jianxiong Xiao. Lsun: Construction of a large-scale image dataset using deep learning with humans in the loop. *arXiv preprint arXiv:1506.03365*, 2015. 4

[134] Bowen Zhang, Shuyang Gu, Bo Zhang, Jianmin Bao, Dong Chen, Fang Wen, Yong Wang, and Baining Guo. Styleswin: Transformer-based gan for high-resolution image generation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11304–11314, 2022. 4

[135] Xu Zhang, Svebor Karaman, and Shih-Fu Chang. Detecting and simulating artifacts in gan fake images. In *2019 IEEE international workshop on information forensics and security (WIFS)*, pages 1–6. IEEE, 2019. 3

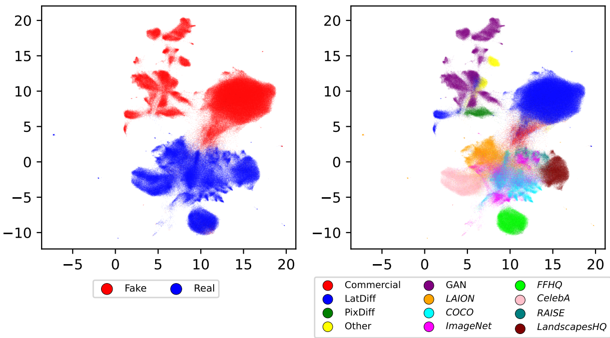
[136] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Computer Vision (ICCV), 2017 IEEE International Conference on*, 2017. 3, 5

[137] Mingjian Zhu, Hanting Chen, Qiangyu YAN, Xudong Huang, Guanyu Lin, Wei Li, Zhijun Tu, Hailin Hu, Jie Hu,

and Yunhe Wang. Genimage: A million-scale benchmark for detecting ai-generated image. In *Advances in Neural Information Processing Systems Dataset and Benchmarks Track*, pages 77771–77782, 2023. 2, 3, 4, 5, 6, 7, 8

[138] Bojia Zi, Minghao Chang, Jingjing Chen, Xingjun Ma, and Yu-Gang Jiang. Wilddeepfake: A challenging real-world dataset for deepfake detection. In *Proceedings of the 28th ACM international conference on multimedia*, 2020. 2

## A. Other applications



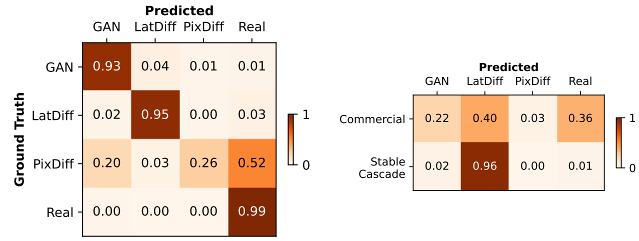

(a) Fake *vs*. real visualization (b) Generator type visualization

Figure 8. **Feature space visualization.** We visualize a feature space of our trained classifier using 10% of our training data and the evaluation set. For better visibility, only a subset of our real datasets are visualized and the labels for real datasets are italicized. We observe a good separation between *fake vs. real* data, and between different generator types and real datasets.

Other applications, beyond the "real-or-fake" image forensics task, could potentially be supported by our dataset. In particular, a diverse array of generators and their corresponding images in our dataset may be valuable for addressing the *generator attribution* problem, where the goal is to identify the characteristics of the underlying generator that is responsible for synthesizing a given image.

Figure 8 presents a UMAP [82] visualization of the feature space of our trained classifier. We use the activation of the penultimate layer for visualization following Ojha *et al*. [90]. The feature space reveals interesting structure: GANs form a clearly separated cluster; most commercial models are distributed closely to latent diffusion models; real datasets such as LAION [111], ImageNet [23], COCO [74], and RAISE [19] are closely distributed, whereas CelebA [76], FFHQ [59], and Landscapes HQ [116] appear to be more isolated. It is important to note that these separations emerge naturally without explicit training. A targeted learning objective may further enhance these separations.

Building on the feature space observations, we use a $k$-nearest-neighbor classifier with $k = 5$ using 10% of our training data to identify the generator types in our evaluation set. We separate generators as "known" (i.e., GANs, latent and pixel diffusions, and real data) and "unknown" (commercial models and Stable Cascade [97]) generator types and compute the confusion matrices as shown in Figure 9. Note that none of these generators are seen during training. Figure 9a demonstrates strong performance in identifying GANs, latent diffusion models, and real data. However, pixel-based diffusion models show lower performance, possibly due to their limited representation (only 3 models) in our training set. The classification result for the "unknown" set is shown in Figure 9b. Interestingly, commercial models are predominantly classified as latent diffusion or GANs, while Stable Cascade [97] displays similarity to latent diffusion models despite their unique three-stage sampling process.




(a) Known architectures (b) Unknown architectures

Figure 9. **Generator type classification.** We classify the generator type of a given image using $k$-nearest-neighbor. (a) Confusion matrix of "known" generator types. We observe high accuracy in GANs, latent diffusions, and real data. (b) Classification results on "unknown" architectures. Commercial models are predominantly classified as latent diffusion and GANs (disregarding 'real'). Stable Cascade [97], which we categorized as *Other* generator type, shows similarity to latent diffusion models.

## B. Dataset composition

**Generator licenses.** In Figure 10, we report the generator licenses in our dataset. Most of the models use the `CreativeML OpenRAIL-M` license [93].

**Model metadata.** We show an example model metadata in Tab. 3. It contains the name of the models, their categorized architectures, licenses, source real datasets, and the Hugging Face tags if available.

**Model composition.** The composition of the training set of Community Forensics is detailed in Table 4 and Fig. 11. A vast majority of the models and generated images are latent diffusion. Figure 12 illustrates the composition of the evaluation set, which includes two variants of HDiT [16]: one trained on FFHQ [59] and another on ImageNet [23]. For computing metrics such as mAP and accuracy, these HDiT variants are treated as separate entities due to their distinct training data and model weights. However, when reporting the number of models in our dataset, we count them as a single model.

## C. Training settings

For training our classifiers, we use `AdamW` optimizer [78] with a learning rate of `2e-5`, a weight decay of `1e-2`, a batch size of `512`, and mixed precision [83]. We use a cosine weight decay with a warmup of 20% of the total iterations. We train our models for `52K` iterations using this setting. For the models in Figures 1 and 4, we employ shorter training iterations (`3K`) due to the computational overhead associated with training a substantial number of models for statistical analysis. We chose this number of iterations since we found

| Model | Architecture | License | RealSource | HF_pipeline_tag | HF_diffusers_tag |
|---|---|---|---|---|---|
| danbochman/ccxl | LatentDiff | None | coco,forchheim,imagenet,imd2020,laion,landscapesHQ,vision | StableDiffusionXL-Pipeline | StableDiffusionXL-Pipeline |
| livingbox/modern-style-v3 | LatentDiff | creativeml-openrail-m | coco,forchheim,imagenet,imd2020,laion,landscapesHQ,vision | StableDiffusion-Pipeline | stable-diffusion |
| ... | | | | | |
| DeepFloyd | PixelDiff | DeepFloyd-IF | coco | N/A | N/A |
| BigGAN | GAN | MIT | imagenet | N/A | N/A |
| ... | | | | | |

Table 3. **Example model metadata.** We log both the author and model names for the Hugging Face [34] models and only the model names for others. We also log the generator type (i.e., architecture), model license, source real dataset, and Hugging Face tags if available.

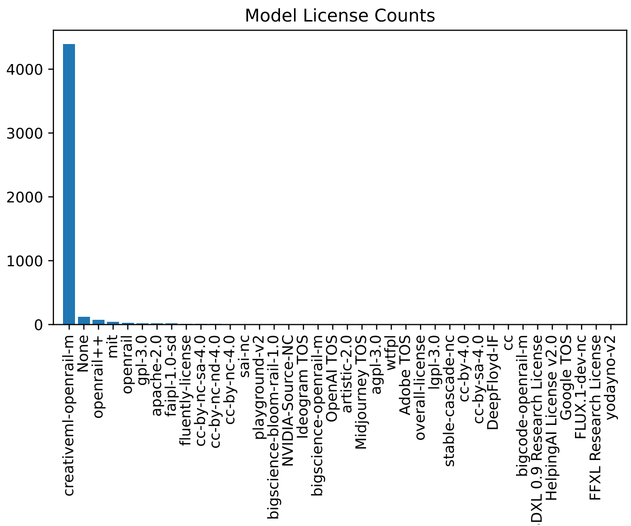



Figure 10. Histogram of model licenses in our dataset. A vast majority of the models use the `CreativeML OpenRAIL-M` license [93].

| | Latent Diff. | GAN | Pixel Diff. | Other |
|---|---|---|---|---|
| Models | 4766 | 12 | 3 | 1 |
| Percentage | 99.67% | 0.25% | 0.06% | 0.02% |

Table 4. Model counts per architecture in the training set. The generators are predominantly latent diffusion models.

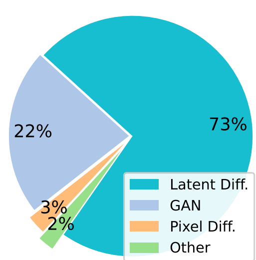



Figure 11. Number of images per generator type in the training set.

that classifier performance begins to plateau with approximately this amount of training (Figure 13).

| | Commercial | Latent Diff. | GAN | Pixel Diff. | Other |
|---|---|---|---|---|---|
| Models | 11 | 6 | 2 | 1 | 1 |
| Images | 14918 | 6000 | 2000 | 2000 | 1000 |

Figure 12. Evaluation set composition.

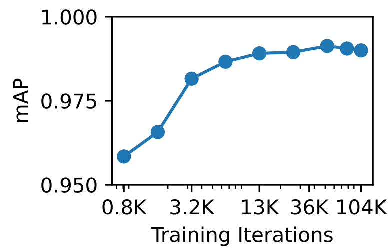



Figure 13. **Impact of training iterations.** The performance of the classifier plateaus beyond 3K iterations.

# D. Example model project page

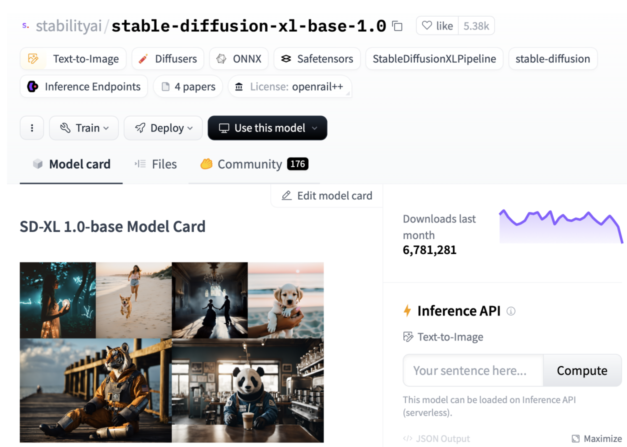



Figure 14. Example model project page from Hugging Face [34].

Figure 14 shows a project page from Hugging Face [34]. We can see the tags associated with the model (e.g., Text-to-image, pipeline type, license), number of downloads, and sample images.