DETECTING GENERATED IMAGES VIA MACHINE UNLEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Robust detection of generated images is critical to counter the misuse of generative models. Existing methods primarily depend on learning from human-annotated training datasets, limiting their generalization to unseen distributions. In contrast, large-scale vision models (LVMs) pre-trained on web-scale datasets exhibit exceptional generalization power through exposure to diverse distributions, offering a transformative paradigm for this task. However, our experimental results reveal that LVMs pre-trained exclusively on natural images effectively capture the features of both natural and generated images to achieve comparably low loss, thereby failing to distinguish both types of images. This prompts a key question: *When and how do LVMs exhibit different behaviors when capturing features of natural and generated images?* This investigation reveals an insight: during unlearning, LVMs exhibit disparate forgetting dynamics with feature degradation for generated images escalating faster than natural ones. Inspired by the disparate dynamics, we introduce two detection methods: 1) data-free detection, which prunes model parameters to induce unlearning without data access, and 2) data-driven detection, which optimizes LVMs to unlearn knowledge tied to generated images. Extensive experiments conducted on various benchmarks demonstrate that our unlearning-based approach outperforms conventional detection methods. By recasting the detection task as a problem of machine unlearning, our work establishes a new paradigm for generated image detection.

1 INTRODUCTION

With the rapid advancements in generative models (Dhariwal & Nichol, 2021; Rombach et al., 2022; Karras et al., 2019), AI-generated images have reached a level of quality that often makes them almost indistinguishable from natural images to the human eye. These developments have unlocked unprecedented potential in areas such as content creation, media, and entertainment, driving innovation across industries. However, the ability to generate hyper-realistic images also introduces significant risks Frank et al. (2020b), especially regarding the potential for misuse in misinformation, privacy invasion, and identity fraud. Consequently, effective and robust detection of AI-generated images has become essential to promote the responsible development and deployment of generative models while protecting users and organizations from malicious use.

Existing methods for detecting AI-generated images focus primarily on learning a boundary between natural and generated images (Wang et al., 2020; Ojha et al., 2023; Tan et al., 2024; Liu et al., 2024b) to construct a binary classifier. In this context, these methods typically collect labeled natural and generated images to construct a large-scale dataset to train binary classifiers, aiming to capture and separate features that uniquely characterize each category. Learning from these collected training images, models can identify subtle distinctions between natural and generated content, yielding impressive detection performance under specific conditions.

Despite their success, these methods face the challenge of domain shifts in two critical aspects, which typically degrades the generalization performance. First, their performance is inherently constrained by the generative models employed to generate training images, potentially leading to generalization failures when encountering images produced by novel generative models. Second, the dependency on natural images introduces cross-domain adaptation challenges, as real-world test environments often contain samples distributionally deviating from training images. Thus, these methods are

usually paired with carefully designed data augmentation techniques such as JPEG to enhance their generalization performance (Wang et al., 2020).

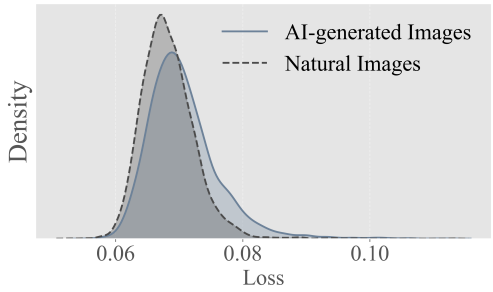


Figure 1: LVMs pre-trained on natural images exhibit low loss for both natural and generated images, thus restricting their discriminative capacity between the types of images.

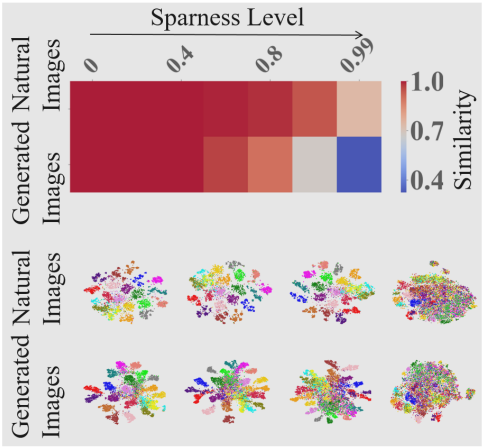


Figure 2: Dynamic illustration of unlearning. As the model transitions from learned to unlearned state, the feature extraction ability of the model shows a clear discrepancy between natural and generated images. We compute feature similarity of images on unlearned model and original model and use t-SNE to visualize the feature distribution of the images on models with different unlearning levels.

Regarding the ability of generalization, pre-trained vision models (LVMs) have shown great success in various domains thanks to their large-scale training with extensive data Mohamed et al. (2022). Thus, LVMs emerge as promising candidates for promoting the detection of generated images. However, as shown in Figure 1, LVMs trained exclusively on natural images have the ability to effectively capture the features of both natural and generated images to achieve comparably low loss. These experimental results imply that employing LVMs fails to detect generated images correctly, which is consistent with the results shown in Figure 1. This prompts a key question: *When and how do LVMs exhibit different behaviors when capturing features of natural and generated images?*

To investigate this fundamental question, we develop a controlled ablation framework to degrade LVM capabilities—a strategy motivated by the inherent difficulty of enhancing LVMs' capabilities, aiming to elicit potential differences in LVMs on two types of images. Specifically, we record and analyze the extent to which the features change as a result of the degraded LVMs' capability. Here, we leverage machine unlearning Bourtoule et al. (2021) to degrade LVMs by pruning a specific proportion of parameters with the smallest absolute values. As shown in Figure 2, our experimental results reveal a distinct pattern: as LVMs unlearn more knowledge, the extracted features of natural and generated images change differently. Namely, LVMs exhibit disparate forgetting dynamics with degradation of extracted features for natural and generated images. For original LVMs, different categories of natural and generated images are well distinguished. However, as they unlearn more knowledge, their capability to extract features from the generated images decays significantly faster than from the natural ones. These disparate forgetting dynamics establish a connection between generated image detection and machine unlearning.

Leveraging this insight, we propose an *unlearning-based* approach to detect generated images. Rather than training a binary classifier from scratch to distinguish between natural and generated images, we investigate whether this distinction can be achieved by unlearning knowledge in pre-trained LVMs. Specifically, we propose a simple data-free unlearning method by weight pruning (Han et al., 2015), which is an effective approach to compress models by removing unimportant parameters. This method is data-free and training-free, holding potential for generalization, as it does not rely on specific types of generative models and natural images. Meanwhile, we propose a data-driven method that unlearns generated images to facilitate the separation of natural and generated images. Experimental results across multiple benchmarks demonstrate that our unlearning-based approach outperforms state-of-the-art methods. Moreover, we further conduct experiments on images generated by inaccessible generative models, i.e., Sora OpenAI (2024), to verify the robustness against domain shifts of our method. Results shown in Table 3 demonstrate that our method consistently and significantly outperforms existing methods.

Our main contributions can be summarized as follows:

- Employing LVMs for generated image detection holds promise in addressing the challenge of domain shift, but our experimental results show that original LVMs exhibit similar loss values for both natural and generated images. To elicit potential differences, we develop a controlled ablation experiment and reveal that degraded LVMs display distinct patterns when extracting features for these two types of images.
- Inspired by the disparate dynamics, we propose two unlearning-based methods to detect generated images: 1) data-free detection inducing unlearning by parameter pruning and 2) data-driven detection optimizing LVMs by unlearning knowledge tied to generated images. This unlearning-based approach shifts the focus from learning boundaries between natural and generated images to unlearning knowledge in pre-trained models.
- Comprehensive experiments validate our method across diverse generated image datasets, demonstrating that our method outperforms existing methods. Moreover, experiments on images generated by inaccessible models verify its robustness against domain shifts.

2 PRELIMINARIES

Given a test image $\mathbf{x}$, the task of AI-generated image detection is to determine whether $\mathbf{x}$ originates from the natural image distribution or is generated by a generative model. A common approach frames this as a supervised binary classification problem, utilizing a training set comprising labeled samples from both distributions, which is formalized as follows.

Let $X^0 = \{\mathbf{x}_1^0, \ldots, \mathbf{x}_{N^0}^0\}$ represent a set of N^0 AI-generated images labeled as 0, and $X^1 = \{\mathbf{x}_1^1, \ldots, \mathbf{x}_{N^1}^1\}$ denote N^1 natural images labeled as 1. The objective is to jointly learn a feature extractor $F(\cdot;\theta_F)$ and a binary classifier $D(\cdot;\theta_D)$, parameterized by θ_F and θ_D, respectively, by minimizing a classification loss $\ell(\cdot)$ over the combined dataset:

$$D, F = \arg\min_{\theta_D, \theta_F} \ell\left(D(F(\mathbf{x};\theta_F);\theta_D), y\right), \tag{1}$$

where $y \in \{0, 1\}$ is the ground-truth label for input $\mathbf{x}$.

Once trained, the model computes a decision score $s(\mathbf{x}) = D(F(\mathbf{x}))$ for each test image $\mathbf{x}$. A hard prediction is obtained by thresholding this score at a fixed value τ:

$$\text{pred}(\mathbf{x}) = \begin{cases} \text{generated}, & \text{if } s(\mathbf{x}) < \tau, \\ \text{natural}, & \text{otherwise.} \end{cases} \tag{2}$$

The robustness of this framework is fundamentally limited by the empirical coverage of the training data. Effective generalization to unseen distributions requires the learned representation $F(\mathbf{x})$ captures features invariant to variations in generative models. However, training sets often provide limited coverage of the diverse generative mechanisms encountered in practice. To enhance the robustness of AI-generated image detectors, prior works (Chen et al., 2024; Zhu et al., 2023a) employ techniques such as data augmentation or adversarial training. Despite these efforts, such methods exhibit limited transferability to samples from unseen generative distributions, highlighting the need for approaches rooted in principled distributional modeling beyond empirical discriminative techniques.

3 METHODOLOGY

3.1 MOTIVATION

Current methods for detecting generated images struggle to generalize to unseen generative distributions. A natural approach to improve generalization is to expand the training dataset, leading to the consideration of large-scale pre-trained models such as DINOv2 (Oquab et al., 2024), which offer robust generalization from extensive pre-training. However, as shown in Figure 1, these models exhibit comparable low loss on both natural and generated images, reflecting their strong comprehension of both domains, which prevents their direct use for discrimination.

To address this, we propose selectively degrading the model's ability to interpret generated images, thus inducing differential performance between natural and generated images. To this end, we employ machine unlearning (Bourtoule et al., 2021), a technique to mitigate the influence of specific data, to adapt the pre-trained model to forget generated images while preserving its representation of natural images. A native unlearning objective function for classification is defined as:

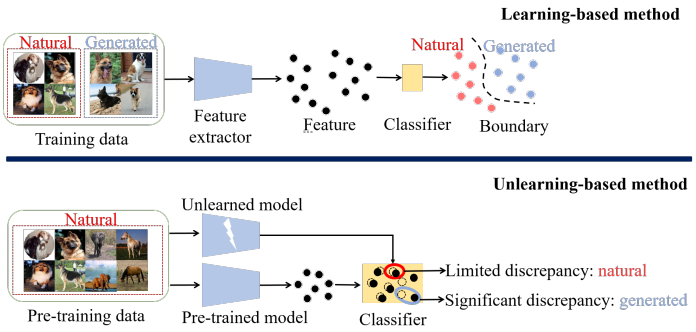


Figure 3: Differences and connections between learning-based detection and unlearning-based detection. Learning-based methods aim to introduce a boundary between natural and generated images, leading to the reliance on the collected training data. In contrast, our unlearning-based method leverages dynamic classifiers for detection.

$$\mathcal{L}_{\text{unlearn}}(\theta) = \mathbb{E}_{(x,y)\sim\mathcal{D}_{\text{forget}}}\left[-\sum_i \frac{1}{K}\log f_i(x;\theta)\right] + \lambda\mathbb{E}_{(x,y)\sim\mathcal{D}_{\text{retain}}}\left[-\sum_i y_i \log f_i(x;\theta)\right], \quad (3)$$

where, in general, $\mathcal{D}_{\text{forget}}$ and $\mathcal{D}_{\text{retain}}$ represent the data distributions to be forgotten and preserved, respectively; in this work, they correspond to generated and natural images.

3.2 Data-free Unlearning

Eq. 3 presents a general machine unlearning framework that requires collecting both natural and generated images to fine-tune the model, incurring additional computational costs. In this study, we investigate strategies to induce selective unlearning in LVMs, prioritizing the retention of natural image knowledge while forgetting generated image knowledge, particularly in scenarios where generated images are unavailable. To do so, we propose a training-free unlearning method by leveraging insights from the pre-training process of large-scale models, achieving effective data-free unlearning. Our idea comes from previous work on the effect of weight pruning on neural networks (Hooker et al., 2019), where the authors found that compression has a greater impact on the long-tail of less frequent instances than on higher frequency instances. On existing pre-trained vision models, such as CLIP (Radford et al., 2021) and DINOv2 (Oquab et al., 2024), most of the pre-training data are natural images, and generated images are rarely used for pre-training, which results in the generated images unintentionally becoming long-tailed, or out-of-distribution (OOD) data. In this case, weight pruning disproportionately affects generated images compared to natural images, effectively achieving unlearning objective for generated images. As illustrated in Figure 4, pruning the weights of DINOv2 results in a significantly larger feature displacement for generated images than for natural images. This phenomenon is consistently observed in Figures 2.

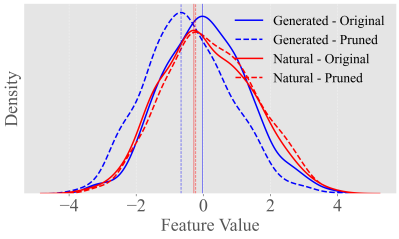


Figure 4: The feature shift caused by weight perturbation is more significant in the generated images.

Based on this, we propose a training-free method for AI-generated image detection, using the feature similarity between a learned model and its unlearned (pruned) version as the criterion:

$$s(x) = \cos(F(x;\theta_F), F(x;\theta'_F)), \quad (4)$$

where cos denotes cosine similarity, and θ'_F is the unlearned version of θ_F.

3.3 Theoretical Analysis of Weight Pruning's Differential Impact

While prior studies (Hooker et al., 2019) have noted the differential impact of weight pruning on in-distribution (ID) and OOD data, they lack theoretical justification. In this work, we provide a theoretical proof to substantiate these observations. To formalize this, we consider a neural network

$f(x;\theta)$ trained solely on natural images, where $x \in \mathcal{X} \subset \mathbb{R}^m$ is the input image, $\theta \in \mathbb{R}^d$ is the weight parameter vector and $f : \mathcal{X} \times \mathbb{R}^d \to \mathbb{R}^k$ is the output feature embeddings. In this case, natural images are considered as ID data and generated images are considered as OOD data.

Definition 1 (Weight Pruning)**.** For a model with parameters $\theta \in \mathbb{R}^d$, weight pruning discards weights with absolute values below a threshold $\epsilon > 0$, defined as:

$$\theta_i' = \begin{cases} \theta_i & \text{if } |\theta_i| \geq \epsilon, \\ 0 & \text{if } |\theta_i| < \epsilon, \end{cases} \tag{5}$$

where θ_i' represents the pruned parameters.

Definition 2 (Generalization Error Increment)**.** For a loss function $\ell(f(x;\theta), y) : \mathbb{R}^k \times \mathcal{Y} \to \mathbb{R}$ measures prediction error, assumed to be Lipschitz continuous with respect to f and twice differentiable with respect to θ. The generalization error for a distribution $\mathcal{D}$ is:

$$\text{Err}_{\mathcal{D}}(f) = \mathbb{E}_{(x,y)\sim\mathcal{D}}[\ell(f(x;\theta), y)]. \tag{6}$$

And for a model $f(x;\theta)$ with pruned parameters θ', resulting in $f' = f(x;\theta')$, the generalization error increment induced by pruning over a distribution $\mathcal{D}$ is defined as:

$$\Delta\text{Err}_{\mathcal{D}} = \text{Err}_{\mathcal{D}}(f') - \text{Err}_{\mathcal{D}}(f) = \mathbb{E}_{(x,y)\sim\mathcal{D}}[\ell(f(x;\theta'), y) - \ell(f(x;\theta), y)]. \tag{7}$$

Specifically, $\Delta\text{Err}_{\text{ID}}$ and $\Delta\text{Err}_{\text{OOD}}$ denote the increments for ID and OOD data, respectively.

We hypothesize that pruning small weights impacts OOD data more significantly, as formalized in the following theorems.

Theorem 3 (Generalization Error Increment)**.** *For a well-trained ReLU network $f(x;\theta)$, assume the loss function $\ell(f(x;\theta), y)$ is twice differentiable with respect to θ and its first-order and second-order derivatives are all Lipschitz-continuous, pruning small weights leads to a larger generalization error increment for OOD data than for ID data:*

$$\Delta Err_{OOD} > \Delta Err_{ID}. \tag{8}$$

Definition 4 (Output Difference)**.** The mean squared output difference due to weight pruning over a distribution $\mathcal{D}$ is defined as:

$$\Delta_{\text{out}}^{\mathcal{D}} = \mathbb{E}_{x\sim\mathcal{D}}[\|f(x;\theta') - f(x;\theta)\|^2], \tag{9}$$

where $f(x;\theta)$ and $f(x;\theta')$ are the model outputs before and after pruning, respectively, and $\|\cdot\|^2$ denotes the squared Euclidean norm.

Corollary 1 (Output Difference)**.** *Under the conditions of Theorem 3, pruning small weights results in a larger output difference for OOD data than for ID data:*

$$\Delta_{out}^{OOD} > \Delta_{out}^{ID}. \tag{10}$$

The proofs, based on the spectral properties of the Hessian and gradient covariance matrices, are provided in Appendix A.4. These results suggest that the feature representations of AI-generated images (OOD) change more significantly after pruning compared to natural images (ID).

3.4 Data-driven Unlearning

While data-free unlearning passively removes generative knowledge through structural degradation, it does not explicitly optimize for forgetting. When generated images are available, we can further introduce a *data-driven unlearning* strategy to guide the unlearning process.

Specifically, for natural images, we encourage the model to retain feature similarity with the original model. For generated images, we enforce a margin-based separation: the model should not produce feature representations too similar to the original model. This is implemented via the following loss:

$$\mathcal{L}(\theta_F') = \mathbb{E}_{x\in X^1}\left[\mathcal{L}_{\text{CE}}(F(x;\theta_F),\ F(x;\theta_F'))\right] + \mathbb{E}_{x\in X^0}\left[\max\left(0,\ \gamma - \mathcal{L}_{\text{CE}}(F(x;\theta_F),\ F(x;\theta_F'))\right)\right], \tag{11}$$

where γ is a margin that enforces a minimal dissimilarity threshold between the original and unlearned features for generated images. The cross-entropy loss $\mathcal{L}_{\text{CE}}(P,Q) = -\sum_i P_i \log Q_i$ measures the divergence between normalized feature distributions extracted by the original model θ_F and the unlearned model θ_F'. After getting the unlearned model, we compute the scoring function using Eq. (4) and make a judgment using Eq. (2).

Table 1: AI-generated image detection performance on ImageNet. Values are percentages. **Bold** numbers are superior results. We compare training methods and training-free methods separately.

Methods	Models																		Average	
	ADM		ADMG		LDM		DiT		BigGAN		GigaGAN		StyleGAN XL		RQ-Transformer		Mask GIT			
	AUROC	AP	AUROC	AP	AUROC	AP	AUROC	AP	AUROC	AP	AUROC	AP	AUROC	AP	AUROC	AP	AUROC	AP	AUROC	AP
Training-free Methods																				
AEROBLADA	55.61	54.26	61.57	56.58	62.67	60.93	85.88	87.71	44.36	45.66	47.39	48.14	47.28	48.54	67.05	67.69	48.05	48.75	57.87	57.85
Data-free Unlearning	**91.97**	**90.44**	**86.82**	**85.14**	**87.62**	**85.91**	**85.74**	**83.84**	**96.37**	**96.52**	**94.39**	**94.23**	**96.47**	**96.53**	**95.19**	**95.24**	**95.27**	**95.17**	**92.20**	**91.45**
Training Methods																				
CNNspot	62.25	63.13	63.28	62.27	63.16	64.81	62.85	61.16	85.71	84.93	74.85	71.45	68.41	68.67	61.83	62.91	60.98	61.69	67.04	66.78
UnivFD	83.37	82.95	79.60	78.15	80.35	79.71	82.93	81.72	93.07	92.77	87.45	84.88	85.36	83.15	85.19	84.22	90.82	90.71	85.35	84.25
DIRE	51.82	50.29	53.14	52.96	52.83	51.84	54.67	55.10	51.62	50.83	50.70	50.27	50.95	51.36	55.95	54.83	52.58	52.10	52.70	52.18
NPR	85.68	80.86	84.34	79.79	91.98	86.96	86.15	81.26	89.73	84.46	82.21	78.20	84.13	78.73	80.21	73.21	89.61	84.15	86.00	80.84
PatchCraft	81.83	79.65	70.88	69.36	68.47	65.19	75.38	73.29	99.85	99.26	98.55	97.91	96.33	96.25	91.28	91.47	92.56	92.17	86.13	84.95
FatFormer	91.77	90.36	83.58	83.17	92.58	92.06	86.93	85.14	98.76	98.47	97.65	98.02	97.64	97.57	96.55	95.96	97.65	97.27	93.68	93.11
DRCT	90.26	90.07	85.74	83.85	90.24	89.88	88.27	89.06	95.87	94.99	86.89	86.12	89.11	88.39	92.38	92.41	94.44	94.47	90.36	89.92
AIDE	90.87	90.17	87.91	85.52	93.57	93.89	89.87	88.16	88.48	88.12	97.93	96.58	96.59	95.97	98.31	97.86	99.87	99.56	93.71	92.87
Data-driven Unlearning	**96.86**	**96.69**	**94.92**	**94.77**	**98.32**	**98.50**	**96.25**	**96.52**	**99.96**	**99.96**	**99.43**	**99.54**	**99.73**	**99.74**	**99.26**	**99.34**	**99.90**	**99.91**	**98.29**	**98.33**

Table 2: Accuracy (%) of different detectors on Chameleon.

Training Set	AEROBLADA	Data-free Unlearning	CNNSpot	FreDect	Fusing	GramNet	LNP	UnivFD	DIRE	NPR	AIDE	DRCT	PatchCraft	FatFormer	Data-driven Unlearning
ProGAN	55.29	**59.17**	56.94	55.62	56.98	58.94	57.11	57.22	58.19	57.29	56.45	57.89	53.76	55.78	**60.59**
SD v1.4	55.29	**59.17**	60.11	56.86	57.07	60.95	55.63	55.62	59.71	58.13	61.10	60.33	56.32	59.34	**71.15**
All GenImage	55.29	**59.17**	60.89	57.22	57.09	59.81	58.52	60.42	57.83	57.81	63.89	61.97	55.70	60.59	**72.89**

Table 3: AI-generated image detection performance, measured by AUROC (%) and AP (%), on Sora.

Models	AEROBLADA		Data-free Unlearning		CNNspot		UnivFD		DIRE		NPR		PatchCraft		FatFormer		DRCT		AIDE		Data-driven Unlearning	
	AUROC	AP	AUROC	AP	AUROC	AP	AUROC	AP	AUROC	AP	AUROC	AP	AUROC	AP	AUROC	AP	AUROC	AP	AUROC	AP	AUROC	AP
Sora	58.00	57.13	**90.89**	**90.18**	52.85	53.29	77.06	80.69	52.83	52.16	51.92	50.25	84.39	82.16	89.95	87.64	82.53	82.28	91.76	89.39	**95.89**	**95.54**
Open Sora	62.37	55.79	**90.00**	**89.10**	50.14	51.38	67.05	68.67	53.66	52.98	50.25	51.84	83.58	81.89	88.76	87.99	81.79	80.11	89.47	88.98	**97.03**	**95.70**
Average	60.19	56.46	**90.45**	**89.64**	51.50	52.84	72.06	74.68	53.25	52.57	51.09	51.05	83.99	82.03	89.36	87.82	82.16	81.20	90.62	89.19	**96.46**	**95.62**

3.5 Relation between learning and unlearning approach

In a sense, our unlearning-based method has the same structure as the learning-based method. In our unlearning method, the feature extractor is instantiated as a pre-trained large-scale vision model, while the weighting of classifiers is instantiated as the output features of the unlearned model on the test sample. Our unlearning methods has two advantages over the learning methods: (1) we directly use the large-scale vision model as the feature extractor, which is pre-trained on a large number of natural images, instead of retraining a feature extractor on a limited number of samples. This allows to obtain more powerful features; and (2) the classifier of the learning approach is fixed once the training is completed; instead, our unlearning method generates an instance-specific classifier for each test sample, which allows the division of the feature space to be independent of the specific natural and generated samples. We illustrate the differences and connections between our unlearning-based method and learning-based approach in Figure 3.

4 Experiments

4.1 Setup

Datasets. Following previous works (Luo et al., 2024; Ojha et al., 2023; Wang et al., 2023; Yan et al., 2024), we conduct extensive comparative experiments on the following benchmarks: **ImageNet** (Deng et al., 2009), **LSUN-BEDROOM** (Yu et al., 2015), **GenImage** (Zhu et al., 2023b), **DiffusionForensics** (Wang et al., 2023), **Chameleon** (Yan et al., 2024) and **DRCT-2M** (Chen et al., 2024). In addition to the public datasets, we evaluate the methods on a proprietary dataset generated using the Sora and OpenSora models.

Implementation Details. For data-free unlearning, we leverage fully parameterized DINOv2 ViT-L/14 as the learned model. It has 24 transformer blocks, and we obtain a sparse model by pruning the parameters of 90% of the minimum magnitude weights of the fc2 layer of its 16th transformer block, and use this model as the unlearned model. We use $1k$ natural images sampled from ImageNet and generated images generated by ProGAN to select hyperparameters. For data-driven unlearning, we leverage LoRa (Hu et al., 2022) for parameter-effcient fine-tuning. The Lora layers are applied on the q_proj and v_proj layers of DINOv2. $lora\_r$ and $lora\_\alpha$ are set to 8. The margin γ is set to 20. When calculating the classification accuracy, the threshold is determined by a set of natural images and generated images. More detailed illustration is provided in Appendix A.12.

Table 4: AI-generated image detection performance on LSUN-BEDROOM.

Methods	ADM		DDPM		iDDPM		Diffusion GAN		Projected GAN		StyleGAN		Unleashing Transformer		Average	
	AUROC	AP	AUROC	AP	AUROC	AP	AUROC	AP	AUROC	AP	AUROC	AP	AUROC	AP	AUROC	AP
Training-free Methods																
AEROBLADA	57.05	58.37	61.57	61.49	59.82	61.06	47.12	48.25	45.98	46.15	45.63	47.06	59.71	57.34	53.85	54.25
Data-free Unlearning	**76.49**	**73.43**	**92.80**	**91.78**	**88.74**	**87.21**	**97.51**	**97.34**	**98.40**	**98.43**	**90.92**	**89.74**	**96.73**	**96.12**	**91.66**	**90.58**
Training Methods																
CNNspot	64.83	64.24	79.04	80.58	76.95	76.28	88.45	87.19	90.80	89.94	95.17	94.94	93.42	93.11	84.09	83.75
UnivFD	71.26	70.95	79.26	78.27	74.80	73.46	84.56	82.91	82.00	78.42	81.22	78.08	83.58	83.48	79.53	77.94
DIRE	57.19	56.85	61.91	61.35	59.82	58.29	53.18	53.48	55.35	54.93	57.66	56.90	67.92	68.33	59.00	58.59
NPR	75.43	72.60	91.42	90.89	89.49	88.25	76.17	74.19	75.07	74.59	68.82	63.53	84.39	83.67	80.11	78.25
DRCT	74.59	71.37	85.45	84.98	87.17	86.99	94.19	94.16	95.96	95.67	93.92	94.66	89.51	89.07	88.68	88.13
Data-driven Unlearning	**89.87**	**90.44**	**99.51**	**99.58**	**99.13**	**99.13**	**99.99**	**99.99**	**99.99**	**99.99**	**99.85**	**99.86**	**99.99**	**99.99**	**98.33**	**98.43**

Table 5: Comparison with linear classification.

Dataset	ImageNet		LSUN-BEDROOM	
Methods	AUROC	AP	AUROC	AP
Linear classification	87.83	86.49	84.72	83.57
Data-free Unlearning	**92.20**	**91.45**	**91.66**	**90.58**

Table 6: Effect of unlearning different parameters.

Metrics	Query	Key	Value	fc1	fc2	all
AUROC	87.55	88.21	90.08	88.19	**92.20**	88.81
AP	85.06	86.02	88.77	87.07	**91.45**	87.13

Evaluation metrics. Following previous works (Ojha et al., 2023; Wang et al., 2023), we take the following metrics: (1) the average precision (AP); (2) the area under the receiver operating characteristic curve (AUROC) and (3) the classification accuracy (ACC).

Baselines. We take the following works as baselines: CNNspot (Wang et al., 2020), UnivFD (Ojha et al., 2023), DIRE (Wang et al., 2023), NPR (Tan et al., 2024), PatchCraft (Zhong et al., 2023), FatFormer (Liu et al., 2024a), DRCT (Chen et al., 2024), AIDE (Yan et al., 2024) and AEROBLADE (Ricker et al., 2024). In addition to the above works, we have also compared our methods on some of benchmarks with the following works: FreDect (Frank et al., 2020a), Fusing (Ju et al., 2022), Durall (Durall et al., 2020), LNP (Liu et al., 2022), F3Net (Qian et al., 2020), SelfBland (Shiohara & Yamasaki, 2022), GANDetection (Mandelli et al., 2022), LGrad (Tan et al., 2023), Spec (Zhang et al., 2019), GenDet (Zhu et al., 2023a), and GramNet (Liu et al., 2020).

4.2 Experimental results

Comparison with other baselines. As shown in Table 1, 2, 13, 12, 4 and 14, we compare our method with other baselines on ImageNet, Chameleon, GenImage, DRCT-2M, LSUN-BEDROOM, and DiffusionForensics, respectively. Results show that our unlearning approach achieves better performance compared to learning approach. Notably, even without any generated images to guide the unlearning process, our simple weight pruning-based unlearning method can achieve good results. And performance is further enhanced when generated images are incorporated to steer the unlearning procedure. To further illustrate the effectiveness of our method, we count the image feature similarity on learned and unlearned models for natural images and generated images, respectively. As shown in Figure 7, the similarity of natural images is significantly higher than that of various generated images, and this difference effectively distinguishes natural images from generated images.

Experimental results on Sora. We further evaluate the performance of our method on unknown video generation models. Specifically, we collect multiple publicly available Sora (OpenAI, 2024) videos and generate multiple videos using Open Sora (Zheng et al., 2024), and sample images from these videos as AI-generated images and sample natural images from Laion-400M (Schuhmann et al., 2021) to evaluate the effectiveness of our method. As shown in Table 3, our method achieves the best performance on the images generated by these unknown generative models. These results highlight the effectiveness of the proposed method.

Table 7: The effect of selecting different pre-trained models.

Model	AUROC	AP
DINOv2: ViT-S/14	74.01	72.86
DINOv2: ViT-B/14	85.74	83.05
DINOv2: ViT-L/14	92.20	91.45
DINOv2: ViT-g/14	88.12	84.73
CLIP: ViT-L/14	85.92	85.65
CLIP: RN50×64	80.03	78.32

4.3 Ablation study

In this section, we perform ablation experiments. Unless otherwise stated, experiments are conducted on ImageNet benchmark.

Robustness to image corruptions. In addition to the performance on clean images, the robustness of the detector to various image corruptions is also an important metric. In reality, images may be

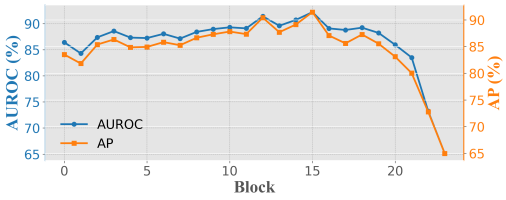


Figure 5: The effects of pruning blocks.

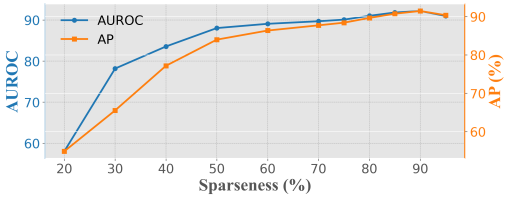
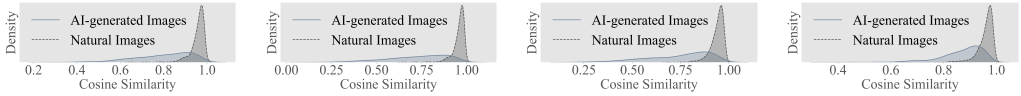


Figure 6: The effect of pruning ratio.



(a) (b) (c) (d)

Figure 7: Comparison of feature similarity on learned and unlearned models. The generated images are from: (a) ADM, (b) BigGAN, and (c) DDPM, and (d) Midjourney.

exposed to various perturbations, e.g., when people upload images to social media, images may be compressed, and these operations may affect the performance of the detector. Following previous work (Wang et al., 2020; Ojha et al., 2023), we assess the robustness of detectors against three types of perturbations: JPEG compression (with quality parameter q)), Gaussian blur (with standard deviation σ), and Gaussian noise (with standard deviation σ). As shown in Figure 8, our method also achieves the best detection performance under different image perturbations.

The effect of model. In our experiments, we focus on using DINOv2 ViT-L/14 as the vision model. To further investigate the effect of vision model, we also test the performance of the unlearning approach on other models. As shown in Table 7, the proposed unlearning method is able to distinguish between natural and generated images across different models.

The effect of γ. Figure 9 illustrates our exploration of how the margin γ influences the performance of our data-driven unlearning approach. The results show that our method is robustness to γ.

The effect of pruning ratio. Figure 6 illustrates how pruning ratio influences the performance of our method. The results show that our method maintains robustness across a wide range of pruning ratio. Performance degradation occurs only when the pruning ratio is too low. At lower pruning ratio, the model unlearns little information and the features extracted on the learned model and unlearned model are almost the same, making unlearning ineffective.

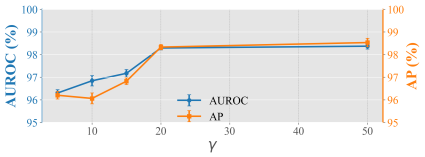
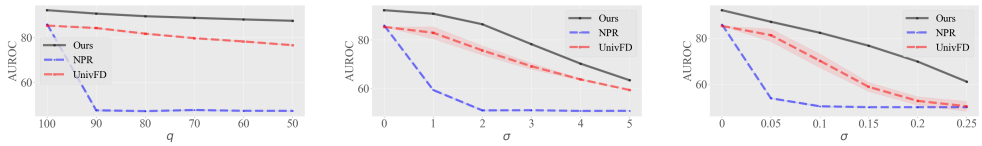


Figure 9: The effect of γ

Select which block for unlearning? As shown in Figure 5, we investigate the impact of selecting different blocks for unlearning on detection performance. The results indicate that our method demonstrates robustness across various blocks. However, when unlearning is applied at the top block, performance experiences a significant degradation. This is likely due to the fact that the top block parameters encode higher-level features, and pruning these parameters directly disrupts the feature representation of natural images. As a result, the feature consistency between the learned and unlearned models for natural images is compromised, leading to a notable decline in the performance.

The effect of parameters. In DinoV2-L/14, a transformer block consists of two main components: the attention module and the Multi-Layer Perceptron (MLP). Among them, the attention module mainly consists of three parameters: Query, Key and Value. MLP mainly consists of two fully connected layers. In our main experiment, we realize unlearning by pruning the parameters of the second fully connected layer (fc2) in the MLP. In Table 6, we further explore the effect of pruning other parameters. The results show that pruning different parameters can obtain good performance.

Comparison with linear classification. We further compare our method to training a single linear layer for binary classification on top of DINOv2-L/14. We use the training set in UnivFD and use JPEG and Blur as data augmentation methods. As shown in Table 5, on the same backbone, our unlearning method also outperforms the learning method.



(a) (b) (c)

Figure 8: Robustness to perturbations: (a) JPEG compression; (b) Gaussian blur; (c) Gaussian noise.

Ablation experiment on LoRa parameters. As shown in Table 8, we conduct ablation experiments on LoRa parameter. The results show that the performance of our unlearning method remains stable under different LoRa parameters, demonstrating the robustness of our method.

5 Related Work

Table 8: The effect of LoRa parameters. We report AUROC/AP.

$lora_\alpha$	$lora\_\gamma = 4$	$lora\_\gamma = 8$	$lora\_\gamma = 16$
8	98.72/98.74	98.29/98.33	98.01/98.20
16	98.48/98.53	98.71/98.79	98.79/98.83
32	98.68/98.74	98.10/98.21	98.61/98.65

AI-generated Image Detection. In recent years, the detection of AI-generated images has emerged as a critical research area, driven by the rapid advancement of generative models Brock et al. (2019); Ho et al. (2020). These models can create hyper-realistic images, which raises concerns around issues such as misinformation, privacy violations, and authenticity. To address these concerns, various methods have been proposed to differentiate between natural and AI-generated images. Early methods in this field (Brock et al., 2019; Ho et al., 2020) largely focus on training specialized binary classifiers to distinguish between natural and generated images. For example, CNNspot (Wang et al., 2020) trains a binary classifier using natural and generated images, where JPEG and Blur are used as data augmentation to improve the robustness of the classifier. UniversalFakeDetect (Ojha et al., 2023) proposes using CLIP's representation space (Radford et al., 2021) to train classifiers, which shows superior performance across a wider range of generative architectures. Gendet (Zhu et al., 2023a) proposes an adversarial teacher-student discrepancy-aware framework, while LaRE^2(Luo et al., 2024) introduces a latent reconstruction error-guided feature refinement approach for detecting images generated by diffusion models. Although these methods have made significant strides, those relying on training still face challenges related to generalization and computational costs. To overcome these limitations, recent studies have shifted focusing toward training-free detection approach. AEROBLADE(Tan et al., 2024) takes a training-free approach by assessing reconstruction errors through autoencoder used in Latent Diffusion Model (LDM) (Rombach et al., 2022), but it is limited to LDM-based generative models. In our paper, the proposed unlearning method is also a train-free method and can be generalized to various generative models.

Machine Unlearning. Machine unlearning has emerged as an important research area due to the need for privacy and security. It focuses on removing the influence of specific data points from a trained model without retraining it from scratch. However, unlearning in deep neural networks is challenging due to their highly non-convex loss functions. (Golatkar et al., 2020) propose a method for scrubbing the weights clean of information about a particular set of training data. (Nguyen et al., 2022) propose a Markov chain Monte Carlo-based machine unlearning algorithm through parameter sampling. These methods balance efficiency and model performance. (Golatkar et al., 2021) propose to set a weight subset to zero to effectively remove all the information contained in the non-core data while minimizing the performance loss. (Jia et al., 2023) utilizes model sparsification via weight pruning to reduce the gap between exact unlearning and approximate unlearning. In this paper, we perform machine unlearning by sparsification.

6 Conclusion

In this paper, we propose a novel machine unlearning framework for detecting AI-generated images. Through rigorous analysis of the differential forgetting dynamics in large-scale vision models during unlearning, we establish that feature degradation for generated images outpaces that for natural images. Leveraging this insight, we develop an unlearning-based detection approach that effectively distinguishes generated images from natural ones. Comprehensive evaluations across diverse benchmarks demonstrate superior performance over existing methods.

REFERENCES

Lucas Bourtoule, Varun Chandrasekaran, Christopher A Choquette-Choo, Hengrui Jia, Adelin Travers, Baiwu Zhang, David Lie, and Nicolas Papernot. Machine unlearning. In *2021 IEEE symposium on security and privacy (SP)*, 2021.

Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale GAN training for high fidelity natural image synthesis. In *International Conference on Learning Representations, ICLR*, 2019.

Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems, NeurIPS*, 2020.

Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *IEEE/CVF International Conference on Computer Vision, ICCV*, 2021.

Baoying Chen, Jishen Zeng, Jianquan Yang, and Rui Yang. DRCT: diffusion reconstruction contrastive training towards universal detection of diffusion generated images. In *International Conference on Machine Learning, ICML*, 2024.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *The IEEE / CVF Computer Vision and Pattern Recognition Conference, CVPR*, 2009.

Prafulla Dhariwal and Alexander Quinn Nichol. Diffusion models beat gans on image synthesis. In *Advances in Neural Information Processing Systems,NeurIPS*, 2021.

Ricard Durall, Margret Keuper, and Janis Keuper. Watch your up-convolution: CNN based generative deep neural networks are failing to reproduce spectral distributions. In *The IEEE / CVF Computer Vision and Pattern Recognition Conference, CVPR*, 2020.

Pierre Foret, Ariel Kleiner, Hossein Mobahi, and Behnam Neyshabur. Sharpness-aware minimization for efficiently improving generalization. In *International Conference on Learning Representations, ICLR*, 2021.

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*, 2020a.

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*, pp. 3247–3258. PMLR, 2020b.

Aditya Golatkar, Alessandro Achille, and Stefano Soatto. Eternal sunshine of the spotless net: Selective forgetting in deep networks. In *The IEEE / CVF Computer Vision and Pattern Recognition Conference, CVPR*, 2020.

Aditya Golatkar, Alessandro Achille, Avinash Ravichandran, Marzia Polito, and Stefano Soatto. Mixed-privacy forgetting in deep networks. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR*, 2021.

Ian Goodfellow, Yoshua Bengio, Aaron Courville, and Yoshua Bengio. *Deep learning*, volume 1. MIT Press, 2016.

Song Han, Jeff Pool, John Tran, and William J. Dally. Learning both weights and connections for efficient neural network. In *Advances in Neural Information Processing Systems,NeurIPS*, 2015.

Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross B. Girshick. Momentum contrast for unsupervised visual representation learning. In *The IEEE / CVF Computer Vision and Pattern Recognition Conference, CVPR*, 2020.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems,NeurIPS*, 2020.

Sepp Hochreiter and Jürgen Schmidhuber. Flat minima. *Neural computation*, 1997.

Sara Hooker, Aaron Courville, Gregory Clark, Yann Dauphin, and Andrea Frome. What do compressed deep neural networks forget? *arXiv preprint arXiv:1911.05248*, 2019.

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. In *International Conference on Learning Representations, ICLR*, 2022.

Jinghan Jia, Jiancheng Liu, Parikshit Ram, Yuguang Yao, Gaowen Liu, Yang Liu, Pranay Sharma, and Sijia Liu. Model sparsity can simplify machine unlearning. In *Advances in Neural Information Processing Systems,NeurIPS*, 2023.

Yan Ju, Shan Jia, Lipeng Ke, Hongfei Xue, Koki Nagano, and Siwei Lyu. Fusing global and local features for generalized ai-synthesized image detection. In *International Conference on Image Processing, ICIP*, 2022.

Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *The IEEE / CVF Computer Vision and Pattern Recognition Conference, CVPR*, 2019.

Bo Liu, Fan Yang, Xiuli Bi, Bin Xiao, Weisheng Li, and Xinbo Gao. Detecting generated images by real images. In *Proceedings of the European Conference on Computer Vision, ECCV*, 2022.

Huan Liu, Zichang Tan, Chuangchuang Tan, Yunchao Wei, Jingdong Wang, and Yao Zhao. Forgery-aware adaptive transformer for generalizable synthetic image detection. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR*, 2024a.

Huan Liu, Zichang Tan, Chuangchuang Tan, Yunchao Wei, Yao Zhao, and Jingdong Wang. Forgery-aware adaptive transformer for generalizable synthetic image detection. In *The IEEE / CVF Computer Vision and Pattern Recognition Conference, CVPR*, 2024b.

Zhengzhe Liu, Xiaojuan Qi, and Philip H. S. Torr. Global texture enhancement for fake face detection in the wild. In *The IEEE / CVF Computer Vision and Pattern Recognition Conference, CVPR*, 2020.

Yunpeng Luo, Junlong Du, Ke Yan, and Shouhong Ding. Lareˆ2: Latent reconstruction error based method for diffusion-generated image detection. In *The IEEE / CVF Computer Vision and Pattern Recognition Conference, CVPR*, 2024.

Sara Mandelli, Nicolò Bonettini, Paolo Bestagini, and Stefano Tubaro. Detecting gan-generated images by orthogonal training of multiple cnns. In *International Conference on Image Processing, ICIP*, 2022.

Abdelrahman Mohamed, Hung-yi Lee, Lasse Borgholt, Jakob D Havtorn, Joakim Edin, Christian Igel, Katrin Kirchhoff, Shang-Wen Li, Karen Livescu, Lars Maaløe, et al. Self-supervised speech representation learning: A review. *IEEE Journal of Selected Topics in Signal Processing*, 16(6): 1179–1210, 2022.

Quoc Phong Nguyen, Ryutaro Oikawa, Dinil Mon Divakaran, Mun Choon Chan, and Bryan Kian Hsiang Low. Markov chain monte carlo-based machine unlearning: Unlearning what needs to be forgotten. In *ASIA CCS '22: ACM Asia Conference on Computer and Communications Security*, 2022.

Utkarsh Ojha, Yuheng Li, and Yong Jae Lee. Towards universal fake image detectors that generalize across generative models. In *The IEEE / CVF Computer Vision and Pattern Recognition Conference, CVPR*, 2023.

OpenAI. Sora: Creating video from text, 2024. URL `https://openai.com/index/sora/`.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Mido Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Hervé Jégou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. Dinov2: Learning robust visual features without supervision. *Trans. Mach. Learn. Res.*, 2024.

Yuyang Qian, Guojun Yin, Lu Sheng, Zixuan Chen, and Jing Shao. Thinking in frequency: Face forgery detection by mining frequency-aware clues. In *Proceedings of the European Conference on Computer Vision, ECCV*, 2020.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning, ICML*, 2021.

Jonas Ricker, Denis Lukovnikov, and Asja Fischer. AEROBLADE: training-free detection of latent diffusion images using autoencoder reconstruction error. In *The IEEE / CVF Computer Vision and Pattern Recognition Conference, CVPR*, 2024.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *The IEEE / CVF Computer Vision and Pattern Recognition Conference, CVPR*, 2022.

Levent Sagun, Utku Evci, V Ugur Guney, Yann Dauphin, and Leon Bottou. Empirical analysis of the hessian of over-parametrized neural networks. *arXiv preprint arXiv:1706.04454*, 2017.

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.

Kaede Shiohara and Toshihiko Yamasaki. Detecting deepfakes with self-blended images. In *The IEEE / CVF Computer Vision and Pattern Recognition Conference, CVPR*, 2022.

George Stein, Jesse C. Cresswell, Rasa Hosseinzadeh, Yi Sui, Brendan Leigh Ross, Valentin Villecroze, Zhaoyan Liu, Anthony L. Caterini, J. Eric T. Taylor, and Gabriel Loaiza-Ganem. Exposing flaws of generative model evaluation metrics and their unfair treatment of diffusion models. In *Advances in Neural Information Processing Systems,NeurIPS*, 2023.

Chuangchuang Tan, Yao Zhao, Shikui Wei, Guanghua Gu, and Yunchao Wei. Learning on gradients: Generalized artifacts representation for gan-generated images detection. In *The IEEE / CVF Computer Vision and Pattern Recognition Conference, CVPR*, 2023.

Chuangchuang Tan, Huan Liu, 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 *The IEEE / CVF Computer Vision and Pattern Recognition Conference, CVPR*, 2024.

Sheng-Yu Wang, Oliver Wang, Richard Zhang, Andrew Owens, and Alexei A. Efros. Cnn-generated images are surprisingly easy to spot... for now. In *The IEEE / CVF Computer Vision and Pattern Recognition Conference, CVPR*, 2020.

Zhendong Wang, Jianmin Bao, Wengang Zhou, Weilun Wang, Hezhen Hu, Hong Chen, and Houqiang Li. DIRE for diffusion-generated image detection. In *IEEE/CVF International Conference on Computer Vision, ICCV*, 2023.

Shilin Yan, Ouxiang Li, Jiayin Cai, Yanbin Hao, Xiaolong Jiang, Yao Hu, and Weidi Xie. A sanity check for ai-generated image detection. *arXiv preprint arXiv:2406.19435*, 2024.

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. *CoRR*, 2015.

Xu Zhang, Svebor Karaman, and Shih-Fu Chang. Detecting and simulating artifacts in GAN fake images. In *International Workshop on Information Forensics and Security, WIFS*, 2019.

Zangwei Zheng, Xiangyu Peng, Tianji Yang, Chenhui Shen, Shenggui Li, Hongxin Liu, Yukun Zhou, Tianyi Li, and Yang You. Open-sora: Democratizing efficient video production for all, March 2024. URL https://github.com/hpcaitech/Open-Sora.

Nan Zhong, Yiran Xu, Zhenxing Qian, and Xinpeng Zhang. Rich and poor texture contrast: A simple yet effective approach for ai-generated image detection. *arXiv preprint arXiv:2311.12397*, 2023.

Mingjian Zhu, Hanting Chen, Mouxiao Huang, Wei Li, Hailin Hu, Jie Hu, and Yunhe Wang. Gendet: Towards good generalizations for ai-generated image detection. *arXiv preprint arXiv:2312.08880*, 2023a.

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,NeurIPS*, 2023b.

A Appendix

A.1 LLM Usage Statement

In the preparation of this manuscript, we utilized Large Language Models (LLMs) to assist with language polishing and refinement of the text. Specifically, the LLM was employed to enhance the clarity, coherence, and grammatical accuracy of the writing, ensuring that the manuscript adheres to high standards of academic communication. The LLM did not contribute to the research ideation, methodology, data analysis, or core content development, which were entirely conducted by the authors. All outputs generated by the LLM were carefully reviewed and edited by the authors to ensure alignment with the intended scientific contributions and to maintain the integrity of the work.

A.2 Social impacts

The proposed method for detecting AI-generated images significantly contributes to mitigating societal risks associated with generative model misuse. By enhancing the capability to identify synthetic media, such as deepfakes, this work bolsters efforts to counter disinformation and fosters trust in digital media, particularly in critical domains such as journalism and legal evidence.

A.3 Limitations

The proposed method relies on a vision foundation model pre-trained only on the natural images. However, with the rapid development of generative models, future models are likely to be contaminated by generative images, which may lead to the failure of our data-free unlearning method.

A.4 Detailed proofs

A.4.1 Proof of Theorem 3 (Generalization Error Increment)

Proof. The generalization error increment for distribution $\mathcal{D}$ is:

$$\Delta\text{Err}_{\mathcal{D}} = \mathbb{E}_{(x,y)\sim\mathcal{D}}[\ell(f(x;\theta'),y) - \ell(f(x;\theta),y)]. \tag{12}$$

Using a second-order Taylor expansion of the loss around θ:

$$\ell(f(x;\theta'),y) = \ell(f(x;\theta),y)+\nabla_\theta\ell(f(x;\theta),y)^\top(\theta'-\theta)+\frac{1}{2}(\theta'-\theta)^\top\nabla^2_\theta\ell(f(x;\theta),y)(\theta'-\theta)+O(\|\theta'-\theta\|^3). \tag{13}$$

Since $\theta' - \theta = -\Delta\theta$, we have:

$$\ell(f(x;\theta'),y) - \ell(f(x;\theta),y) \approx -\nabla_\theta\ell(f(x;\theta),y)^\top\Delta\theta + \frac{1}{2}\Delta\theta^\top\nabla^2_\theta\ell(f(x;\theta),y)\Delta\theta. \tag{14}$$

Taking the expectation over $\mathcal{D}$:

$$\Delta\text{Err}_{\mathcal{D}} \approx -\mathbb{E}_{(x,y)\sim\mathcal{D}}[\nabla_\theta \ell(f(x;\theta),y)^\top \Delta\theta] + \frac{1}{2}\mathbb{E}_{(x,y)\sim\mathcal{D}}[\Delta\theta^\top \nabla_\theta^2 \ell(f(x;\theta),y)\Delta\theta]. \tag{15}$$

Define the Hessian:

$$H_{\mathcal{D}} = \mathbb{E}_{(x,y)\sim\mathcal{D}}[\nabla_\theta^2 \ell(f(x;\theta),y)]. \tag{16}$$

Thus:

$$\mathbb{E}_{(x,y)\sim\mathcal{D}}[\Delta\theta^\top \nabla_\theta^2 \ell(f(x;\theta),y)\Delta\theta] = \Delta\theta^\top H_{\mathcal{D}}\Delta\theta. \tag{17}$$

For the first-order term:

$$\mathbb{E}_{(x,y)\sim\mathcal{D}}[\nabla_\theta \ell(f(x;\theta),y)] = \nabla_\theta \mathbb{E}_{(x,y)\sim\mathcal{D}}[\ell(f(x;\theta),y)]. \tag{18}$$

Since $\Delta\theta$ corresponds to small weights, the perturbation is small and training minimizes the gradient magnitude in these directions (Sagun et al., 2017), so the first-order term is negligible and thus:

$$\Delta\text{Err}_{\mathcal{D}} \approx \frac{1}{2}\Delta\theta^\top H_{\mathcal{D}}\Delta\theta. \tag{19}$$

Specifically:

$$\Delta\text{Err}_{\text{ID}} \approx \frac{1}{2}\Delta\theta^\top H_{\text{ID}}\Delta\theta, \tag{20}$$

$$\Delta\text{Err}_{\text{OOD}} \approx \frac{1}{2}\Delta\theta^\top H_{\text{OOD}}\Delta\theta. \tag{21}$$

Since $\Delta\theta$ represents small-magnitude weights, these weights are usually not important for ID data, i.e., H_{OOD} has larger eigenvalues in the direction $\Delta\theta$ compared to H_{ID}. Represent the Hessian difference:

$$H_{\text{OOD}} = H_{\text{ID}} + A, \tag{22}$$

where $A = H_{\text{OOD}} - H_{\text{ID}}$ is a symmetric matrix. Generally speaking, training on $\mathcal{D}_{\text{ID}}$ flattens the loss landscape in small weight directions (smaller H_{ID} eigenvalues) (Hochreiter & Schmidhuber, 1997; Foret et al., 2021), while OOD data retains higher curvature due to distributional differences. We assume A has positive eigenvalues in the direction $\Delta\theta$, i.e., $\Delta\theta^\top A\Delta\theta > 0$, therefore:

$$\Delta\theta^\top H_{\text{OOD}}\Delta\theta = \Delta\theta^\top H_{\text{ID}}\Delta\theta + \Delta\theta^\top A\Delta\theta > \Delta\theta^\top H_{\text{ID}}\Delta\theta. \tag{23}$$

Thus:

$$\Delta\text{Err}_{\text{OOD}} \approx \frac{1}{2}\Delta\theta^\top H_{\text{OOD}}\Delta\theta > \frac{1}{2}\Delta\theta^\top H_{\text{ID}}\Delta\theta \approx \Delta\text{Err}_{\text{ID}}. \tag{24}$$

This completes the proof. □

A.4.2 Proof of Corollary 1 (Output Difference)

Proof. Under the conditions of Theorem 3, we relate the generalization error increment to the output difference. The Hessian of the loss with respect to the parameters is: $H_{\mathcal{D}}$ is:

$$H_{\mathcal{D}} = \mathbb{E}_{(x,y)\sim\mathcal{D}}\left[\nabla_\theta f(x;\theta)^\top \frac{\partial^2 \ell}{\partial f^2}\nabla_\theta f(x;\theta) + \frac{\partial \ell}{\partial f}\nabla_\theta^2 f(x;\theta)\right]. \tag{25}$$

where $f(x;\theta)$ is the network output, $\nabla_\theta f(x;\theta)$ is the Jacobian, and $\nabla_\theta^2 f(x;\theta)$ is the second-order derivative tensor. For a ReLU network, $f(x;\theta)$ is piecewise linear, so $\nabla_\theta^2 f(x;\theta) = 0$ for inputs x where the activation pattern is fixed (i.e., no ReLU threshold crossings). And since the first-order of loss function is Lipschitz-continuous, the seceond term is negligible:

$$\mathbb{E}_{(x,y)\sim\mathcal{D}}\left[\frac{\partial \ell}{\partial f}\nabla_\theta^2 f(x;\theta)\right] \approx 0. \tag{26}$$

The Hessian simplifies to:

$$H_{\mathcal{D}} \approx \mathbb{E}_{(x,y)\sim\mathcal{D}}\left[\nabla_\theta f(x;\theta)^\top \frac{\partial^2 \ell}{\partial f^2}\nabla_\theta f(x;\theta)\right]. \tag{27}$$

For the loss function such as cross-entropy loss, the second derivative $\frac{\partial^2 \ell}{\partial f^2}$ is approximately diagonal near the optimum, as the softmax output stabilizes. Assume:

$$\frac{\partial^2 \ell}{\partial f^2} \approx C_{\mathcal{D}} I, \tag{28}$$

where $C_{\mathcal{D}} > 0$ is a positive constant reflecting the loss's curvature in the output space (e.g., related to the inverse of the softmax temperature). This approximation holds when the model's predictions are confident, as shown in (Goodfellow et al., 2016). Define the covariance matrix of the parameter gradients:

$$\Sigma_{\mathcal{D}} = \mathbb{E}_{(x,y)\sim\mathcal{D}}\left[\nabla_\theta f(x;\theta)\nabla_\theta f(x;\theta)^\top\right]. \tag{29}$$

Thus:

$$H_{\mathcal{D}} \approx C_{\mathcal{D}}\Sigma_{\mathcal{D}}, \tag{30}$$

where $\Sigma_{\mathcal{D}} \in \mathbb{R}^{n\times n}$ is positive semi-definite, with eigenvalues reflecting the variability of the network's output sensitivity to parameter changes.

Define the output difference as the expected squared change in the network's output:

$$\Delta_{\text{out}}^{\mathcal{D}} = \mathbb{E}_{(x,y)\sim\mathcal{D}}[\|f(x;\theta') - f(x;\theta)\|_2^2]. \tag{31}$$

For small perturbations, approximate $f(x;\theta') \approx f(x;\theta) - \nabla_\theta f(x;\theta)^\top \Delta\theta$, so:

$$f(x;\theta') - f(x;\theta) \approx -\nabla_\theta f(x;\theta)^\top \Delta\theta. \tag{32}$$

Thus:

$$\Delta_{\text{out}}^{\mathcal{D}} \approx \mathbb{E}_{(x,y)\sim\mathcal{D}}[\|\nabla_\theta f(x;\theta)^\top \Delta\theta\|_2^2] = \mathbb{E}_{(x,y)\sim\mathcal{D}}[\Delta\theta^\top \nabla_\theta f(x;\theta)\nabla_\theta f(x;\theta)^\top \Delta\theta] = \Delta\theta^\top \Sigma_{\mathcal{D}} \Delta\theta. \tag{33}$$

Therefore:

$$\Delta\text{Err}_{\mathcal{D}} \approx \frac{1}{2} C_{\mathcal{D}} \Delta_{\text{out}}^{\mathcal{D}}. \tag{34}$$

$C_{\mathcal{D}}$ only depends on the output value of the network. Since the generated images are very similar to natural images, we assume that the neural network has consistent output distributions on the two types of data, which can be confirmed from Figure 1. Therefore: $C_{\text{OOD}} \approx C_{\text{ID}}$.

Accoring to Theorem 3 and Eq. 34, we have:

$$\Delta_{\text{out}}^{\text{OOD}} > \Delta_{\text{out}}^{\text{ID}}.$$

This completes the proof.

□

A.5 ALLEVIATE SENSITIVITY TO MODELS THROUGH FINE-TUNING

As shown in Table 7, data-free Unlearning exhibits sensitivity to the choice of feature extractor, as it depends on subtle distinctions in how natural and generated images are represented. For the CLIP model, which is trained with text supervision, extracted image features tend to prioritize semantic information, potentially reducing their suitability for this task. Conversely, the DINOv2:ViT-g/14 model, with its large parameter count, shows limited sensitivity to pruning parameters from a single layer, resulting in minimal impact on the final feature representations. This sensitivity to the backbone can be alleviated through fine-tuning the feature extractor, as demonstrated in the results presented in Table 9.

Table 9: Alleviate sensitivity to models through fine-tunings.

Method	Model	AUROC	AP
Data-free Unlearning	DINOv2:ViT-L/14	92.20	91.45
Data-free Unlearning	DINOv2:ViT-g/14	88.12	84.73
Data-free Unlearning	CLIP:ViT-L/14	85.92	85.65
Data-driven Unlearning	DINOv2:ViT-L/14	98.29	98.33
Data-driven Unlearning	DINOv2:ViT-g/14	97.96	98.15
Data-driven Unlearning	CLIP:ViT-L/14	97.63	97.48

A.6 PERFORMANCE OF DATA-FREE UNLEARNING ON WEAKER MODELS

We conduct additional experiments using weaker pretrained vision models, including MoCo (He et al., 2020), SwAV (Caron et al., 2020), and DINO (Caron et al., 2021). As presented in Table 10, our Data-free Unlearning exhibit significantly reduced performance on these models. This highlights a promising direction to explore detection with small models.

Table 10: Performance on weak models.

Models	AUROC	AP
MoCo	72.69	70.15
SwAV	77.85	75.64
DINO	74.79	71.88

A.7 COMPARISON OF COMPUTATIONAL EFFICIENCY

As shown in Table 11, we conduct a comparative analysis of the training and inference costs of various methods on the ImageNet dataset. For inference cost assessment, we measured the time required to detect 100 images. Experimental results show that our unlearning method also shows advantages in computational cost.

Table 11: Comparison of computational efficiency.

Methods	training cost (huors)	inference cost (seconds)
UnivFD	0.8	1.1
NPR	0.7	0.7
DRCT	25.4	1.1
AEROBLADA	0.0	17.6
Data-free Unlearning	0.0	2.5
Data-driven Unlearning	1.5	2.5

A.8 UNLEARNING WITH DIFFERENT PRUNING STRATEGY

In our experiments, we focus on unlearning by pruning the weights with the smallest magnitude. We further explore ablation experiments by randomly pruning the weights, pruning the weights with the

largest magnitude, and filling in the smallest magnitude weights with Gaussian noise instead of using 0 after pruning. As shown in Table 15, simply pruning the weights with the smallest magnitude for unlearning achieves the best performance.

A.9 EXPERIMENTAL RESULTS ON DRCT-2M, GENIMAGE AND DIFFUSIONFORENSICS

Table 12, 13 and 14 shows the performance of our unlearning approach on DRCT-2M, GenImage and DiffusionForensics, respectively. The results further demonstrate the effectiveness of our unlearning approach.

Table 12: AI-generated image detection performance (ACC, %) on DRCT-2M.

Method	SD Variants						Turbo Variants		LCM Variants		ControlNet Variants			DR Variants			Avg.
	LDM	SDv1.4	SDv1.5	SDv2	SDXL	SDXL-Refiner	SD-Turbo	SDXL-Turbo	LCM-SDv1.5	LCM-SDXL	SDv1-Ctrl	SDv2-Ctrl	SDXL-Ctrl	SDv1-DR	SDv2-DR	SDXL-DR	
CNNSpot	99.87	99.91	99.90	97.63	66.25	86.55	86.15	72.42	98.26	61.72	97.96	85.89	82.94	60.93	51.41	50.28	81.12
F3Net	99.85	99.78	99.79	88.60	55.85	87.37	63.29	63.66	97.39	54.98	97.98	72.39	81.99	65.42	50.39	50.27	71.13
CLIP/RN50	99.00	99.99	99.96	94.61	62.08	91.43	84.40	64.40	98.97	57.43	99.74	80.69	82.03	65.83	50.67	50.47	80.05
GramNet	99.40	99.01	98.84	95.30	62.63	80.68	71.19	69.32	93.05	57.02	89.97	75.55	82.68	51.23	50.01	50.08	76.62
De-fake	92.1	95.53	99.51	89.65	64.02	69.24	92.00	93.93	99.13	70.89	58.98	62.34	66.66	50.12	50.16	50.00	75.52
Conv-B	**99.97**	**100.0**	**99.97**	95.84	64.44	82.00	60.75	99.27	99.27	62.33	**99.80**	83.40	73.28	61.65	51.79	50.41	79.11
UniFD	98.30	96.22	96.33	93.83	91.01	93.91	86.38	85.92	90.44	89.99	90.41	81.06	89.06	51.96	51.03	50.46	83.46
FatFormer	96.52	95.31	93.27	91.99	92.87	91.78	88.15	87.48	92.82	91.76	90.28	86.99	88.19	65.92	60.15	55.13	85.53
DIRE	54.62	75.89	76.04	**99.87**	59.90	93.08	97.55	87.29	72.53	67.85	99.69	64.40	64.40	49.96	52.48	49.92	72.55
DRCT	94.45	94.35	94.24	95.05	96.41	95.38	94.81	94.48	91.66	95.54	93.86	93.50	93.54	**84.34**	**83.20**	67.61	91.35
Data-free Unlearning	93.87	72.41	71.82	77.64	83.23	75.39	71.58	67.59	66.84	80.67	84.12	83.89	88.93	70.67	69.14	68.59	76.69
Data-driven Unlearning	98.73	98.93	99.23	99.55	**98.90**	**99.44**	**99.32**	**99.30**	**99.33**	**99.02**	99.14	**99.29**	**98.87**	76.83	74.63	**73.65**	**94.50**

Table 13: AI-generated image detection performance (ACC, %) on GenImage.

Methods	Midjourney	SD V1.4	SD V1.5	ADM	GLIDE	Wukong	VQDM	BigGAN	Average
Training-free Methods									
AEROBLADE	**80.3**	**87.5**	**86.8**	67.2	**81.5**	**83.7**	51.1	52.5	73.8
Data-free Unlearning	79.9	80.8	79.6	**77.7**	79.1	82.7	**87.1**	**87.3**	**81.8**
Training Methods									
ResNet-50	54.9	99.9	99.7	53.5	61.9	98.2	56.6	52.0	72.1
DeiT-S	55.6	99.9	99.8	49.8	58.1	98.9	56.9	53.5	71.6
Swin-T	62.1	99.9	99.8	49.8	67.6	99.1	62.3	57.6	74.8
CNNspot	52.8	96.3	95.9	50.1	39.8	78.6	53.4	46.8	64.2
Spec	52.0	99.4	99.2	49.7	49.8	94.8	55.6	49.8	68.8
F3Net	50.1	99.9	**99.9**	49.9	50.0	**99.9**	49.9	49.9	68.7
GramNet	54.2	99.2	99.1	50.3	54.6	98.9	50.8	51.7	69.9
DIRE	60.2	99.9	99.8	50.9	55.0	99.2	50.1	50.2	70.7
UnivFD	73.2	84.2	84.0	55.2	76.9	75.6	56.9	80.3	73.3
PatchCraft	79.0	89.5	89.3	77.3	78.4	89.3	83.7	72.4	82.3
NPR	81.0	98.2	97.9	76.9	89.8	96.9	84.1	84.2	88.6
FatFormer	92.7	**100.0**	**99.9**	75.9	88.0	99.9	98.8	55.8	88.9
GenDet	89.6	96.1	96.1	58.0	78.4	92.8	66.5	75.0	81.6
DRCT	**91.5**	95.0	94.4	**79.4**	89.1	94.6	90.0	81.6	89.4
AIDE	79.4	99.7	99.8	78.5	**91.8**	98.7	80.3	66.9	86.9
Data-driven Unlearning	90.8	95.6	95.1	74.5	90.0	94.6	91.8	**85.1**	**89.7**

Table 14: AI-generated image detection performance (ACC, %) on DiffusionForensics.

Methods	Models																Average	
	ADM		DDPM		iDDPM		LDM		PNDM		VQ-Diffusion		SDV1		SDV2			
	ACC	AP	ACC	AP	ACC	AP	ACC	AP	ACC	AP	ACC	AP	ACC	AP	ACC	AP	ACC	AP
CNNspot	53.9	71.8	62.7	76.6	50.2	82.7	50.4	78.7	50.8	90.3	50.0	71.0	38.0	76.7	52.0	90.3	51.0	79.8
UnivFD	78.4	92.1	72.9	78.8	75.0	92.8	82.2	97.1	75.3	92.5	83.5	97.7	56.4	90.4	71.5	92.4	74.4	91.7
Frank	58.9	65.9	37.0	27.6	51.4	65.0	51.7	48.5	44.0	38.2	51.7	66.7	32.8	52.3	40.8	37.5	46.0	50.2
Durall	39.8	42.1	52.9	49.8	55.3	56.7	43.1	39.9	44.5	47.3	38.6	38.3	39.5	56.3	62.1	55.8	47.0	48.3
SelfBland	57.0	59.0	61.9	49.6	63.2	66.9	83.3	92.2	48.2	48.2	77.2	82.7	46.2	68.0	71.2	73.9	63.5	67.6
GANDetection	51.1	53.1	62.3	46.4	50.2	63.0	51.6	48.1	50.6	79.0	51.1	51.2	39.8	65.6	50.1	36.9	50.8	55.4
Patchfor	77.5	**93.9**	62.3	97.1	50.0	91.6	**99.5**	**100.0**	50.2	99.9	**100.0**	**100.0**	90.7	**99.8**	**94.8**	**100.0**	78.1	97.8
Data-free Unlearning	79.8	85.0	87.5	94.3	88.3	95.3	80.2	89.5	94.2	98.7	92.3	98.0	**93.1**	97.9	92.8	97.9	88.5	94.6
Data-driven Unlearning	**86.9**	93.4	**98.5**	**99.9**	**98.4**	**99.9**	95.4	99.0	**99.1**	**100.0**	99.2	**100.0**	92.2	97.5	93.5	98.1	**95.4**	**98.5**

A.10 THE EFFECT OF STRUCTURED PRUNING

In our main experiment, we explore the effect of unstructured pruning, i.e., removing some of the weights in certain blocks individually. We further explore the effect of structured pruning, i.e., removing a certain block completely. As shown in Figure 10, removing shallow blocks usually gives stable results, whereas removing top blocks results in a more significant impact on the features of

the natural image due to their closer connection to the output features. Thus, this results in poor detection performance. An exception is that when the second block is removed, the proposed method is completely unable to distinguish between natural images and AI-generated images. This may stem from the fact that the second block in DINOv2 is crucial for the extraction of the image features, and when the second block is removed, the model is unable to correctly extract the features of test images.

Table 15: The effect of pruning strategy.

Model	AUROC	AP
random pruning	86.29	85.54
pruning largest magnitude weights	83.69	83.21
pruning smallest magnitude weights	92.20	91.45
pruning and filling noise	86.03	85.64

A.11 Using the similarity of the middle layer features as the decision score

As shown in Table 16, we further explore the effect of using the feature similarity of other middle layers as decision scores. Since we prune the weights of block 15, the outputs of the test samples on the learned and unlearned models differ from blocks 16 to block 23. Therefore, we explore the effect of feature similarity using the output of blocks from block 16 to block 23. The results show that using high-level features to compute the similarity could achieve good results. This is because there is a significant difference between the high-level features of the natural image and the generated image on the learned and unlearned models.

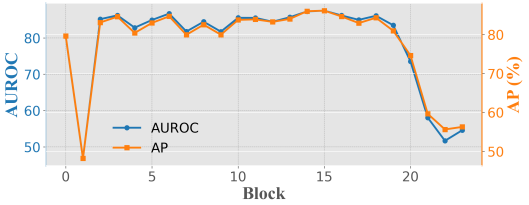


Figure 10: The effect of structured pruning. We obtain an unlearned model by completely dropping a full block.

A.12 Implementation Details

For data-free unlearning, we leverage fully parameterized DINOv2 ViT-L/14 as the learned model. It has 24 transformer blocks, and we obtain a sparse model by pruning the parameters of 90% of the minimum magnitude weights of the fc2 layer of its 16th transformer block, and use this model as the unlearned model. We use $1k$ natural images sampled from ImageNet and generated images generated by ProGAN to select hyperparameters. For data-driven unlearning, we leverage LoRa (Hu et al., 2022) for parameter-effcient fine-tuning. The Lora layers are applied on the q_proj and v_proj layers of DINOv2. $lora\_r$ and $lora\_\alpha$ are set to 8. The margin γ is set to 20. To optimize computational efficiency, we apply Low-Rank Adaptation (LoRA) exclusively to the 18th, 19th, and 20th blocks of the model and fine-tune for only three epochs. The model is optimized using the AdamW optimizer with a learning rate of 1×10^{-5}, $\beta_1 = 0.9$, $\beta_2 = 0.99$, and a weight decay of 0.01. We report the average results under five different random seeds and report the standard deviation in Figure 9. Following CNNspot (Wang et al., 2020), data augmentation techniques including JPEG compression and Gaussian blur are employed to enhance robustness. For the IMAGENNET, LSUN-BEDROOM, and DiffusionForensics benchmarks, the ProGAN dataset serves as the training set. For the GenImage benchmark, SDv1.4 dataset is used. For the DRCT-2M benchmark, SDv2 dataset is used as training set.

When comparing classification accuracy with other methods, since our method is not a standard binary classifier, the traditional classification threshold of 0.5 is not applicable to our method. Consequently, we employed a validation set to determine an appropriate threshold. Specifically, this validation set consisted of 1,000 images generated by ProGAN and an equivalent number of natural images. We identified the threshold that maximized classification accuracy on this validation set as the optimal threshold for subsequent analyses. The determined optimal thresholds were 0.94287 for the data-free unlearning method and 0.90178 for the data-driven unlearning method, with classification accuracy calculated accordingly at these thresholds.

Table 16: Effectiveness of using feature similarity in the middle layer for detection.

Block	Generative Models																		Average	
	ADM		ADMG		LDM		DiT		BigGAN		GigaGAN		StyleGAN XL		RQ-Transformer		Mask GIT			
	AUROC	AP	AUROC	AP	AUROC	AP	AUROC	AP	AUROC	AP	AUROC	AP	AUROC	AP	AUROC	AP	AUROC	AP	AUROC	AP
16	70.32	69.20	65.32	63.28	82.62	83.88	85.69	86.05	88.20	89.79	76.42	77.61	82.59	83.33	81.48	83.96	86.64	88.82	79.92	80.66
17	63.93	65.25	61.22	61.21	72.37	74.71	80.19	82.40	84.65	87.70	61.02	62.82	66.01	66.85	73.34	76.30	80.59	83.72	71.48	73.77
18	78.64	76.76	73.38	70.61	85.27	84.43	88.34	88.05	94.47	94.37	83.34	83.75	88.27	88.82	84.97	84.02	94.23	94.51	85.66	85.04
19	86.90	86.57	81.04	78.92	88.03	88.01	89.80	89.71	96.51	96.69	90.86	91.23	93.55	93.99	89.77	90.06	96.54	96.72	90.33	90.21
20	86.42	86.31	81.82	79.37	89.43	89.60	89.72	89.36	96.89	96.99	91.50	91.57	94.77	94.96	91.56	91.74	96.18	96.30	90.92	90.69
21	87.15	87.57	82.93	80.84	90.00	90.33	89.99	90.36	96.18	96.42	92.08	92.55	95.21	95.52	92.57	93.28	95.80	96.08	91.33	91.44
22	91.04	89.61	85.53	82.90	88.00	86.60	86.37	84.51	96.29	96.71	94.51	94.58	96.65	96.80	94.90	95.22	95.97	96.19	92.14	91.46
23	91.97	90.44	86.82	85.14	87.62	85.91	85.74	83.84	96.37	96.52	94.39	94.23	96.47	96.53	95.19	95.24	95.27	95.17	92.20	91.45

A.13 DETAILS OF DATASETS

ImageNet and LSUN-BEDROOM. The natural images and AI-generated images of ImageNet benchmark and LSUN-BEDROOM benchmark can be obtained from `https://github.com/layer6ai-labs/dgm-eval`, which are provided by (Stein et al., 2023). The generated images of the ImageNet benchmark are generated with the following generative models: ADM, ADMG, BigGAN, DiT-XL-2, GigaGAN, LDM, StyleGAN-XL, RQ-Transformer, and Mask-GIT. The generated images of the LSUN-BEDROOM benchmark are generated with the following generative models: ADM, DDPM, iDDPM, StyleGAN, Diffusion-Projected GAN, Projected GAN, and Unleashing Transformers.

GenImage. The natural images and AI-generated images can be obtained from `https://github.com/GenImage-Dataset/GenImage`. The images are provided by (Zhu et al., 2023b). The generative model includes Midjourney, SD V1.4, SD V1.5, ADM, GLIDE, Wukong, VQDM, and BigGAN. The natural images come from ImageNet.

Chameleon. Chameleon is a a very challenging dataset and various detection methods perform unsatisfactorily on it, as all AI-generated images in this dataset have passed a human perception "Turing Test", i.e., human annotators have misclassified them as natural images. The images are provided by (Yan et al., 2024). The dataset can be obtained from `https://shilinyan99.github.io/AIDE/`.

DiffusionForensics. The natural images and AI-generated images of DiffusionForensics can be obtained from `https://github.com/ZhendongWang6/DIRE`, which are provided by (Wang et al., 2023). The generative model includes ADM, DDPM, iDDPM, LDM, PNDM, VQ-Diffusion, sdv1 and sdv2.

DRCT-2M. The natural images of DRCT-2M come from CoCo and can be obtained from `https://cocodataset.org/#download`. AI-generated images of DRCT-2M can be obtained from `https://modelscope.cn/datasets/BokingChen/DRCT-2M/files`, which are provided by (Chen et al., 2024). The generative model includes LDM, SDv1.4, SDv1.5, SDv2, SDXL, SDXL-Refiner, SD-Turbo, SDXL-Turbo, LCM-SDv1.5, LCM-SDXL, SDv1-Ctrl, SDv2-Ctrl, SDXL-Ctrl, SDv1-DR, SDv2-DR, SDXL-DR.