Adaptive Test-Time Semantic Debiasing for AI-Generated Image Detection

Yu Cai[1] Jiahe Tian[2] Xiaomeng Fu[2] Jiao Dai[2] Jizhong Han[2]
Siwei Lyu[1]
[1]State University of New York at Buffalo, USA
[2]Institute of Information Engineering, Chinese Academy of Sciences, China

Abstract

AI-generated image detectors have historically concentrated on generalization across generative models, often overlooking the critical challenge of cross-semantic generalizability. This limitation constrains the adaptability of detectors to new semantic content in real-world settings. We propose Adaptive Test-Time Semantic Debiasing (ATTSD), a zero-shot approach that utilizes the visual-semantic space of large pretrained vision-language models to dynamically align feature representations during testing—without requiring additional training data or annotations. To further enhance adaptability, we introduce Semantic-Suppression for hard sample mining, adjusting the degree of semantic debiasing for each sample based on Fourier transform properties. To assess cross-semantic generalizability, we present the Cross-Semantic AI-generated Image Detection dataset (CSAIID), a benchmark comprising diverse semantic categories reflective of real-world complexities. Extensive experiments show that ATTSD achieves state-of-the-art performance, particularly excelling in cross-semantic scenarios, positioning it as a promising solution for detecting evolving AI-generated content. The CSAIID dataset is publicly available here.

1. Introduction

As generative AI technologies have advanced in recent years, AI-generated visual content is flooding the online information ecosystem at an unprecedented rate. This influx has posed significant challenges to the trustworthiness of online media, cybersecurity, and copyright protection [1, 37]. Detecting AI-generated images (AGIs) has become an important research topic [18, 28]. From generative adversarial networks (GANs) [2, 13] to the recent diffusion models (DMs) [6, 23], the rapid evolution of image generators poses ongoing challenges for AGI detection. As a result, mainstream detection paradigms have focused their efforts on improving cross-generator generalizability.

A common strategy is to train detectors on images generated by a particular model (*e.g.*, ProGAN [13]) and then evaluate them on images from various unseen generators [18, 28, 32]. However, this strategy mainly focuses on generalization relative to different generators, neglecting another essential factor: the impact of image semantics on detector performance.

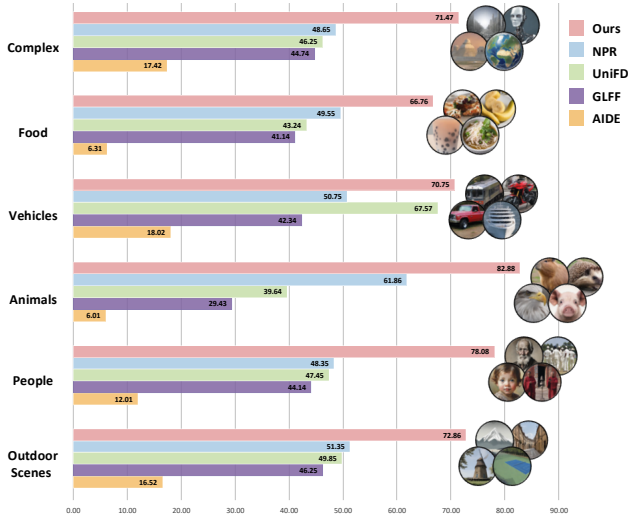

Figure 1. Comparison of detection accuracy on forgeries across various semantic categories. Existing detectors demonstrate limited cross-semantic generalizability. In contrast, our method achieves superior detection accuracy. Note, these detectors are trained on 20 single-object categories from the training set [29] (*e.g.*, horse, cat). The test set, sampled from LCM subset of the cross-semantic dataset proposed in this work, is organized into broader categories. For example, "Animals" subset includes a diverse range of animals such as bears, cats, and birds, reflecting more complex and varied semantics than isolated categories in the training set.

Our analysis demonstrates that image semantics are crucial to a detector's generalizability. Prior research has not comprehensively investigated generalization across varying image semantics, particularly when detectors encounter images with unseen semantic content absent from the training

set. Can an AGI detector trained on specific semantic categories still identify semantically distinct images?

As shown in Fig. 1, we evaluate the performance of existing AGI detectors on images across diverse semantic content. Most detectors show limited effectiveness, achieving only moderate accuracy on specific semantic subsets (*e.g.*, Vehicles and Animals), likely due to the presence of objects similar to those in the training set. This highlights a spurious correlation in current detection formulations, which tend to overfit specific semantic bias present in the training set. While expanding the diversity of training data could potentially address this issue, it is impractical to encompass all possible semantic variations in open-world scenarios.

In this work, we introduce a new perspective on generalizable AGI detection: *How can a detector trained on a closed-set dataset adapt dynamically to emerging, unseen semantic content?* To mitigate the semantic bias between test and training data, we propose performing adaptive semantic debiasing at test time. This approach must satisfy two key requirements: (i) employ unbiased and reliable semantic representations for test samples, and (ii) function in a self-supervised manner, as manually labeling semantic annotations for newly encountered samples is both labor-intensive and potentially introduces human bias.

To this end, we propose visual-semantic alignment as an auxiliary task, built upon the visual space of large-scale pretrained vision-language models. As an instantiation of this idea, we utilize the visual features of CLIP [22], which fulfills aforementioned requirements through two merits: (i) Pretrained on internet-scale datasets through tasks unrelated to forgery, CLIP provides a rich and less biased visual-semantic representation, ideal for semantic alignment; (ii) Each sample can thus be self-labeled by its CLIP visual embeddings, eliminating the need for additional annotations.

Based on this, we propose our Test-Time Semantic Debiasing (TTSD) strategy via a multitask framework where AGI detection serves as the main task and visual-semantic alignment operates as an auxiliary task. Inspired by test-time training [26], we organize these two tasks in a two-branch architecture with a sharing a common feature extractor as their foundation. Upon encountering new test data, the visual-semantic alignment is conducted by distilling the CLIP visual embeddings through the pathway of the auxiliary task. Thereafter, the semantic features learned by the detector are adjusted as the shared feature extractor dynamically adapts to the test distribution. Furthermore, we introduce a Semantic-Suppression strategy for hard sample mining that operates in conjunction with TTSD. This adaptive component adjusts the degree of semantic debiasing applied to each individual sample, forming our complete Adaptive TTSD (ATTSD) method.

We also introduce a new benchmark specifically designed to evaluate the cross-semantic generalizability of AGI detectors. Existing benchmarks rarely consider the semantic dimension. When they do, they typically organize datasets into narrowly defined, single-object categories, such as "car" or "cat," overlooking more complex scenarios and multi-object compositions [29, 38]. In contrast, our Cross-Semantic AI-generated Image Detection dataset (CSAIID) categorizes images under broader conceptual groups, capturing diverse objects and scenarios to better reflect real-world semantic diversity.

Our main contributions are as follows: (1) We identify and address the overlooked challenge of cross-semantic generalizability in the current AGI detection field, revealing the impact of semantic diversity on the generalization; (2) We propose an Adaptive Test-Time Semantic Debiasing (ATTSD) strategy, enabling detectors to dynamically adjust feature learning according to evolving semantic distributions. Our ATTSD retains the zero-shot generalization setting as it requires no additional training data or annotations; (3) We develop the CSAIID benchmark, specifically designed to assess cross-semantic generalizability, and organize it into multiple high-level semantic categories encompassing diverse and multi-object scenarios to reflect real-world visual complexity; (4) Extensive experiments on multiple datasets demonstrate that our approach achieves state-of-the-art generalization performance, particularly in cross-semantic detection scenarios.

2. Related work

2.1. AI-generated image detection

Existing AGI detection methods can be categorized into two types: artifact-based methods and semantic-based methods. Artifact-based methods capture inherent forgery artifacts left by generative models [4, 29, 30, 33]. CNNSpot [29] simulates the artifacts of generative models to detect various GAN-generated fakes. NPR [28] identifies common up-sampling artifacts present in both GAN and DM forgeries. DIRE [30] and DRCT [4] use diffusion-based image reconstruction to detect DM-generated fakes. In contrast, semantic-based methods aim to avoid overfitting to low-level artifacts. UniFD [18] uses the fixed CLIP visual space through linear probing to learn common properties of fake images. De-fake [24] combines image and caption inputs into CLIP's visual and text encoders to detect images created with text-to-image generative models. AIDE [32] proposes to improve detection generalization by combining ConvNeXt-based CLIP semantic features with noise patterns. However, none of the above methods have examined AGI detection from a cross-semantic perspective.

2.2. Test-Time Training

Test-time training (TTT) is a methodology initially designed to enhance a target task by incorporating a self-

supervised auxiliary task during the testing phase, enabling the model to dynamically adapt to distribution shifts between training and test data. TTT is primarily derived from object recognition tasks. Sun et al. designed a rotation prediction task as an auxiliary task to update model weights and help the main task adapt to test domains. TTT-MAE [8] trains a masked autoencoder to reconstruct test images, improving object recognition under distribution shifts. Due to its flexibility and effectiveness, TTT has been introduced to more mainstream tasks. TPT [25] and DiffTPT [7] enhance vision-language models with test-time prompt tuning, generating augmented views of test samples via AugMix [11] or diffusion-based augmentation tools to ensure prediction consistency with the original views. The primary assumption in these works is that applying strong augmentations to test samples will not significantly alter the predictions from the original, as these are all semantic-related tasks. Both reconstruction and diffusion-based generation are semantic-invariant augmentations, as the high-level semantic patterns (*i.e.*, task-specific patterns) remain preserved. However, this strategy may not be directly applicable to AGI detection tasks, as aggressive augmentations could disrupt the low-level artifact patterns that are critical for classification. Given the inverse nature of task-specific patterns, a forgery-invariant augmentation would be more ideal for TTT-involved AGI detection tasks.

3. Methodology

3.1. Preliminaries

To establish notation, consider a standard AGI detector composed of a feature extractor, θ_{ext}, and a forgery classifier, θ_{cls}. The vector of stacked parameters $\boldsymbol{\theta} = (\theta_{ext}, \theta_{cls})$ specifies the entire detection model. Following TTT terminology, we refer to this as the main task. We assume access to a training dataset: $\{(x^1, y^1_{det}), \ldots, (x^n, y^n_{det})\}$. y_{det} denotes the binary real/fake label. The training process can be formulated via binary cross entropy (BCE) loss as follows:

$$\min_{\theta_{ext},\theta_{cls}} \frac{1}{n} \sum_{i=1}^{n} \mathcal{L}_{bce} \left(x^i, y^i_{det}; \theta_{ext}, \theta_{cls}\right). \tag{1}$$

3.2. Visual-semantic alignment

Following the TTT approach, our ATTSD-based detection employs a self-supervised auxiliary task of visual-semantic alignment, *i.e.*, distilling the CLIP visual features. CLIP visual features have proven effective across various visual-language tasks. Given its training on large-scale datasets, CLIP's visual space offers rich semantic representation and is less biased toward specific semantics. Another advantage is that, instead of relying on hard semantic representations such as image captions or numeric class labels, we can directly use the dense representations output by CLIP's visual encoder as soft labels. This manner covers more comprehensive visual content without losing any indescribable visual patterns. We use the original CLIP embeddings without any dimensionality reduction or projection, preserving the integrity of CLIP's semantic representation.

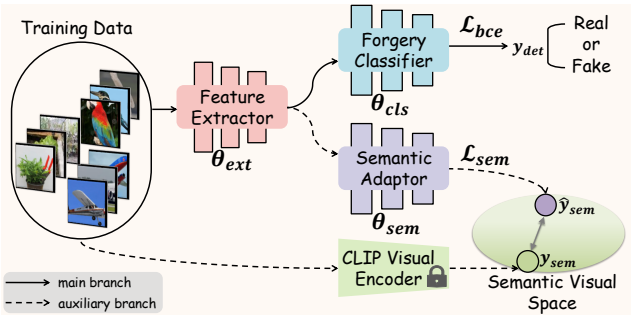

Figure 2. The training scheme of multitask learning for ATTSD. The main task is AGI detection, utilizing the branch composed of the Forgery Classifier and Feature Extractor. The Forgery Classifier predicts the binary real/fake label y_{det}, optimized using the BCE loss $\mathcal{L}_{bce}$. The auxiliary task, visual-semantic alignment, is designed for subsequent test-time training. It operates on the shared Feature Extractor and aligns output features with CLIP's visual semantic space via the Semantic Adaptor, optimized using the semantic alignment loss $\mathcal{L}_{sem}$.

The auxiliary task shares the feature extractor with the main task but requires its task-specific parameters. To facilitate this, we embed a semantic adaptor, θ_{sem}, between the feature extractor and the forgery classifier. Pictorially, this joint architecture forms a Y-shaped model with a shared base and two branches. Following typical TTT work [26], the auxiliary branch mirrors the architecture of the main branch, except for the output dimensionality in the final layer to account for the differing ground truth label between the two tasks. Given CLIP visual embeddings, y_{sem}, as soft labels for the auxiliary task, the visual-semantic alignment performs distillation of the CLIP visual space, thereby updating the feature extractor through the semantic adaptor. This process calibrates the detector's visual features to an unbiased semantic space, as illustrated in Fig. 2.

Training is conducted in a multitask learning fashion on the same training data. The losses for each task are combined, and the gradients are calculated using all the parameters. The joint training objective is therefore formulated as follows:

$$\min_{\theta_{ext},\theta_{sem},\theta_{cls}} \frac{1}{n} \sum_{i=1}^{n} \left[\begin{array}{l} \mathcal{L}_{bce} \left(x^i, y^i_{det}; \theta_{ext}, \theta_{cls}\right) \\ +\mathcal{L}_{sem} \left(x^i, y^i_{sem}; \theta_{ext}, \theta_{sem}\right) \end{array} \right], \tag{2}$$

where $\mathcal{L}_{sem}$ is the semantic alignment loss, aligning the output of the semantic adaptor, $\hat{y}_{sem}$, with the soft semantic label, y_{sem}, by maximizing their cosine similarity.

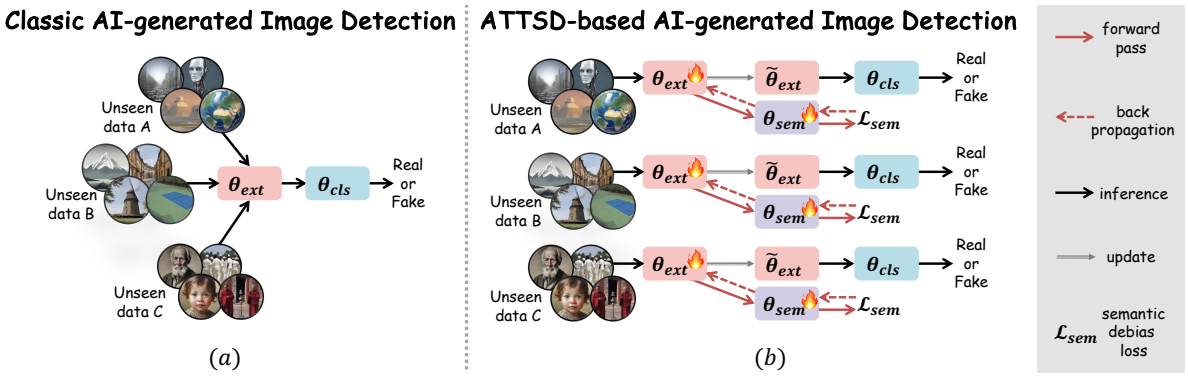

Figure 3. Comparison of test procedures between Classic AI-generated image (AGI) Detection (a) and ATTSD-based AGI Detection (b). In the classic detection procedure, the model remains fixed when encountering unseen data. In contrast, ATTSD-based detection dynamically updates the feature extractor, θ_{ext}, via the semantic adaptor, θ_{sem}, for new test samples. Our detection paradigm allows the model's features to align with evolving semantic distributions in a self-supervised manner, leading to improved detection accuracy.

3.3. Adaptive Test-Time Semantic Debiasing

As shown in Fig. 3, we aim to dynamically update the detector during the test phase to adapt to new test data. In the following, we first describe the test phase of our TTSD-based detection.

Benefiting from the self-supervised visual-semantic alignment, the auxiliary task operates independently of the main task during testing, enabling a zero-shot generalization setting for the main task. During the test, the forgery classifier keeps frozen. TTSD is achieved by finetuning the shared feature extractor towards the objective of the auxiliary task on test images, thereby adapting the model to evolving test distributions.:

$$\min_{\theta_{ext},\theta_{sem}} \frac{1}{n}\sum_{i=1}^{n} \mathcal{L}_{sem}\left(x^i, y^i_{sem}; \theta_{ext}, \theta_{sem}\right). \tag{3}$$

We keep the semantic adaptor tuning during the test phase, as empirical evidence shows that freezing it results in degraded performance (see discussion in Sec. 5.4). Freezing the semantic adaptor forces the feature extractor to adjust excessively to meet the auxiliary task objectives, which can cause it to forget the knowledge required for the main task. Therefore, we maintain the tuning setup throughout all of our experiments.

Let $\widetilde{\theta}_{ext}$ denote the (approximate) minimizer of Eq. (3). Inference is then completed by making a prediction using the updated feature extractor and the original forgery classifier, defined as $\boldsymbol{\theta}(x) = \theta_{cls}\left(\widetilde{\theta}_{ext}(x)\right)$.

Semantic-Suppression for hard sample mining. Semantic bias manifests differently across individual samples, necessitating a sample-specific approach to debiasing. To address this heterogeneity, we introduce a Semantic-Suppression strategy that dynamically modulates the intensity of TTSD. This mechanism assigns adaptive weights to each test sample, precisely adjusting the degree of semantic debiasing applied to its feature representations. The integration of this adaptive component with TTSD yields our complete framework, Adaptive TTSD (ATTSD), which optimizes debiasing on a per-sample basis.

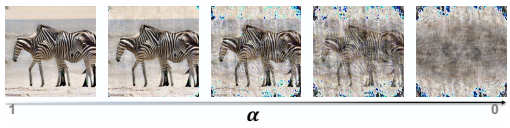

Figure 4. Visualization of the Semantic-Suppression effect by reducing the phase component. As the scaling factor α decreases, the semantic structure of the image progressively deteriorates.

We aim to create a forgery-invariant augmented version for each sample to evaluate its hardness. Inspired by a well-known property of Fourier transformation of an image: the phase component, $\mathcal{P}(x)$, preserves the high-level semantics of the original signal, while the amplitude component, $\mathcal{A}(x)$, contains low-level statistics [19, 20, 31]. We suppress image semantics by applying a modified inverse Fourier transform:

$$x_{aug} = \mathcal{F}^{-1}\left(\mathcal{A}(x) \cdot e^{-j\cdot\alpha\cdot\mathcal{P}(x)}\right), \tag{4}$$

where $\mathcal{F}^{-1}$ denotes operator of inverse Fourier transform. This modified transform generates an augmented version of x, denoted as x_{aug}, where high-level semantics are suppressed by applying a scaling factor $\alpha \in \{0, 1\}$ on the phase component while keeping the amplitude component intact. Thereby, the image semantics are suppressed while retaining low-level statistics crucial for forgery-related patterns to the greatest extent possible (see samples in Fig. 4). The prediction discrepancy $\mathcal{D}$ between the original and augmented image is computed to inform the adaptive weight, with more significant discrepancies indicating a need for more intensive semantic debiasing. This measure underlies our hard

sample mining strategy, where an adaptive weight $\boldsymbol{w}$ is defined for each sample as follows:

$$\boldsymbol{\mathcal{D}} = \|\boldsymbol{\theta}(x) - \boldsymbol{\theta}(x_{aug})\|_2, \tag{5}$$

$$\boldsymbol{w} = \sigma(\boldsymbol{\mathcal{D}}), \tag{6}$$

where σ represents the sigmoid function for normalization. The final ATTSD objective function, integrating the Semantic Suppression strategy, is given by:

$$\min_{\theta_{ext},\theta_{sem}} \frac{1}{n}\sum_{i=1}^{n} \boldsymbol{w^i}\mathcal{L}_{sem}\left(x^i, y^i_{sem}; \theta_{ext}, \theta_{sem}\right). \tag{7}$$

4. Cross-Semantic AI-generated Image Detection Dataset

We now describe the construction of our CSAIID dataset. To maximize semantic coverage, we utilize the COCO dataset [14] as our real image source due to its complex scenes and inclusion of multiple objects. For synthetic images, we employ two complementary approaches. First, we use the original captions from the COCO validation set to generate corresponding fakes. Since current image forgery techniques can not only manipulate real scenarios but also generate entirely fictional content, we further incorporate the PartiPrompts dataset [35], which contains over 1,600 diverse prompts spanning various categories, such as abstract concepts, world knowledge, and fantastical illustrations.

Combining the original annotations of COCO and categories from PartiPrompts, we summarize five dominant semantic categories reflecting real-world scenarios: *"Outdoor Scenes", "People", "Animals", "Vehicles", "Food"*, along with an additional *"Complex"* category, to organize images into corresponding subsets. The *"Complex"* category includes images with intricate scenes that defy clear categorization, as well as fakes generated by prompts from some intricate categories, such as abstract concepts, world knowledge, and fantastical illustrations. Since synthetic images generated from PartiPrompts lack corresponding real counterparts, we sample real images from the COCO training set within the same category to maintain balance.

To facilitate semantic control, all synthetic images are created using text-to-image models. We synthesize images using three generators: Stable Diffusion v2 (SDv2) [23], Latent Diffusion Model (LDM) [23], and Latent Consistency Model (LCM) [16], ensuring a balanced 1:1:1 distribution of images from each generator across subsets. Each subset consists of 1,000 real images and 1,000 synthetic images, resulting in a total dataset size of 12,000 images.

5. Experiments

5.1. Settings

Datasets: Besides our proposed CSAIID dataset, we also use two other widely-used benchmark datasets, UniForen [18] and GenImage [38], to evaluate our method. To optimize the test-time setup of ATTSD before practical evaluation on target test data, we introduce DF3 [12] dataset as a validation set. Its moderate scale allows for an efficient assessment of each component's contribution.

• **UniForen [18].** This dataset extends the GAN-based dataset in [29] by incorporating DM-based images. The training set consists of fake images generated by ProGAN and real images from the LSUN [34] dataset, with each image featuring a single object from 20 classes (*e.g.*, horse and cat). The test set comprises 19 subsets generated by various models, including both GAN and DM generators.

• **GenImage [38].** This dataset mainly focuses on DMs-based fake images using ImageNet [5] as the real source. It includes fakes generated by eight different models. Following the experimental setup of GenImage, we use SDv1.4 as training set and evaluate the test subset of each generator.

• **DF3 [12].** This face-centered dataset contains six forgery generators. It is used as the validation set due to its nonoverlapping image semantics with the training sets of existing benchmarks, satisfying both cross-generator and cross-semantic requirements for evaluating generalizability.

Implementation details: The detector is a simplified ResNet50 model [10], utilizing only the first two stages of the network. The blocks from the first stage act as the feature extractor, while the second stage, combined with a fully connected (FC) layer, serves as the forgery classifier. The semantic adaptor shares the same structure as the forgery classifier. We employ the visual encoder of CLIP: ViT-L/14 to extract semantic embeddings. Following [27], we preprocess input images by extracting NPR-based upsampling artifacts before feeding them into the detector. The α in Eq. (4) is set to 0.9 to minimize excessive disruption to the image. We use the Adam optimizer with a learning rate of 0.0002 and a batch size of 32 for training. During the test phase, the learning rate is 0.0001. Following the standard TTT setup [26], we set batch size as 1 during test. We perform six gradient steps for each sample to adapt the model. These parameters were selected through empirical experiments on the DF3 dataset to balance computational efficiency and adaptation performance.

5.2. Cross-semantic generalization

We first evaluate the performance of state-of-the-art open-source detection models on our proposed CSAIID dataset. To highlight the impact of training data semantics, we group the results based on the source data of the training sets, *i.e.*, LSUN serves as the real source for the UniForen dataset, while ImageNet provides the source for the GenImage dataset. As the creation of fake images inherently simulates the content of their source images, the choice of source data strongly shapes the semantic distribution of each detector learned during training.

Method	Real Source	Forgery Model	Outdoor Scenes	People	Animals	Vehicles	Food	Complex	Avg. Acc.(%)
GLFF [12]	LSUN	PG	63.03	61.68	55.88	63.93	61.58	62.00	61.35
NoDown [9]	LSUN	PG	52.53	51.08	50.28	53.18	51.03	53.10	51.87
NoDown-S [9]	LSUN	SG	51.78	52.88	51.88	51.63	50.73	51.30	51.70
UniFD [18]	LSUN	PG	64.28	64.73	62.38	72.19	61.73	63.35	64.78
DIRE [30]	LSUN	SG	50.23	49.87	49.67	50.08	50.03	49.45	49.89
AIDE [32]	LSUN	PG	53.83	52.08	53.88	54.88	50.78	52.70	53.03
NPR* [28]	LSUN	PG	71.64	73.54	66.68	71.09	**70.89**	74.15	71.33
Ours	LSUN	PG	**76.59**	**82.74**	**87.64**	**80.94**	**81.89**	**84.00**	**82.30**
DIRE [30]	ImageNet	ADM	50.03	49.52	48.82	49.57	49.52	49.15	49.44
AIDE [32]	ImageNet	SD1.4	71.04	69.93	74.79	72.79	77.34	74.45	73.38
NPR* [28]	ImageNet	SD1.4	**86.19**	79.79	80.74	87.09	86.04	84.30	84.02
DRCT [4]	ImageNet	SD1.4	63.42	63.78	64.48	63.63	58.68	64.05	62.85
Ours	ImageNet	SD1.4	86.14	**90.75**	**92.90**	**90.70**	**91.25**	**90.75**	**90.42**

Table 1. Accuracy (ACC, %) comparison of our method with existing generated image detectors across various semantic subsets on the proposed CSAIID dataset. Results are grouped based on different training sources. Except for NoDown-S and DIRE, all methods with each group are trained on the same training set. *i.e.*, the training sets of the UniForen and GenImage benchmarks, respectively. The results of previous detectors are reproduced using official provided weights, while * denotes our re-implemented training with the official codes. "Real Source" and "Forgery Model" specify the real data and forgery model used for training, and "PG" and "SG" denote abbreviations for ProGAN and StyleGAN, respectively.

Method	Generative Adversarial Networks						Deep fakes	Low level vision		Perceptual loss		Guided	LDM			Glide			DALL-E	Total
	Pro-GAN	Cycle-GAN	Big-GAN	Style-GAN	Gau-GAN	Star-GAN		SITD	SAN	CRN	IMLE		200 steps	200 w/ CFG	100 steps	100 27	50 27	100 10		Avg. Acc.(%)
CNNSpot [29]	99.99	85.20	70.20	85.70	78.95	91.70	53.47	66.67	48.69	86.31	86.26	60.07	54.03	54.96	54.14	60.78	63.80	65.66	55.58	69.58
Patchfor [3]	75.03	68.97	68.47	79.16	64.23	63.94	75.54	75.14	75.28	72.33	55.30	67.41	76.50	76.10	75.77	74.81	73.28	68.52	67.91	71.24
NoDown* [9]	100.00	89.97	93.10	97.08	94.27	50.83	98.77	51.67	70.00	94.11	94.12	91.30	60.80	64.00	61.30	66.35	71.15	67.15	60.50	77.71
Co-occurence [17]	97.70	63.15	53.75	92.50	51.10	54.70	57.10	63.06	55.85	65.65	65.80	60.50	70.70	70.55	71.00	70.25	69.60	69.90	67.55	66.86
Spec [36]	49.90	99.90	50.50	49.90	50.30	99.70	50.10	50.00	48.00	50.60	50.10	50.90	50.40	50.40	50.30	51.70	51.40	50.40	50.00	55.45
AIDE* [32]	99.99	98.48	83.95	99.64	73.25	99.91	52.21	64.17	50.91	80.05	86.74	53.10	81.35	62.90	82.00	64.20	65.20	64.20	71.30	75.45
UniFD [18]	100.0	98.50	94.50	82.00	99.50	97.00	66.60	63.00	57.50	59.50	72.00	70.03	94.19	73.76	94.36	79.07	79.85	78.14	86.78	81.38
NPR* [28]	99.99	86.83	87.70	96.90	84.49	99.52	66.88	64.44	70.09	50.00	50.00	74.65	90.90	93.15	92.20	95.20	95.95	95.90	87.95	83.30
Ours	99.80	98.64	93.75	97.86	98.03	100.00	74.26	50.83	53.88	98.12	99.15	89.50	99.20	99.15	99.60	99.45	99.45	99.80	99.60	**92.11**

Table 2. Accuracy (ACC, %) comparison of cross-generator generalization on the UniForen dataset. All methods were trained on the ProGAN subset and evaluated on subsets from different generators. * indicates results obtained using the re-implemented or official pretrained model. The results of the remaining methods are cited from UniFD [18].

As shown in Tab. 1, our method outperforms existing approaches under both training settings, demonstrating superior generalizability in cross-semantic scenarios. Notably, when the training set transitions from LSUN-based to ImageNet-based sources, many methods exhibit significant improvements. For example, AIDE achieves a notable gain of 20.27%, NPR improves by 12.69%, and our method achieves an increase of 8.12%. These improvements arise because ImageNet's broader semantic coverage mitigates the overfitting relative to LSUN's narrow scope. The comparatively smaller gain of our method suggests it is inherently less sensitive to shifts in the semantic distribution of training data. Although these improvements are partially influenced by changes in generators (examined in following subsection), they reaffirm our earlier observations: semantic factors play a crucial role in generalizable detection.

The overall results highlight the semantic limitations commonly associated with closed-set training. Nevertheless, our ATTSD demonstrates substantial improvements across both training sources, suggesting its capability to better adapt to diverse semantic content.

5.3. Cross-generator generalization

We examine cross-generator generalization on two benchmarks: UniForen and GenImage. According to the generators of their training sets, generalization can be assessed from GAN-based and DM-based generators, respectively.

Generalization from GAN-based generators. The models were trained on UniForen's ProGAN-based training set. The results across different generators are presented in Tab. 2. The overall performance of previous methods reveals that generalizing across different generative model

Method	Midjourney	SDv1.4	SDv1.5	ADM	GLIDE	Wukong	VQDM	BigGAN	Avg.
CNNSpot [29]	84.92	99.88	99.76	53.48	53.80	99.68	55.50	49.93	74.62
F3Net [21]	77.85	98.99	99.08	51.20	54.87	97.92	58.99	49.21	73.51
GramNet [15]	73.68	98.85	98.79	51.52	55.38	95.38	55.15	49.41	72.27
De-fake [24]	79.88	98.65	98.62	71.57	78.05	98.42	78.31	74.37	84.73
UniFD [18]	91.46	96.41	96.14	58.07	73.40	94.53	67.83	57.72	79.45
NPR* [28]	79.03	99.52	99.48	62.69	93.18	97.79	55.80	61.97	81.18
DIRE [30]	50.40	99.99	99.92	52.32	67.23	99.98	50.10	49.99	71.24
AIDE [32]	79.38	99.74	99.76	78.54	91.82	98.65	80.26	66.89	86.88
DRCT [4]	91.50	95.01	94.41	79.42	89.18	94.67	90.03	81.67	89.49
Ours	91.77	94.72	93.82	84.98	97.34	91.44	81.22	90.18	**90.68**

Table 3. Accuracy (ACC, %) comparison of cross-generator generalization on the GenImage dataset. All methods were trained on the SDv1.4 training set and evaluated on subsets from different generators. * indicates results obtained using the re-implemented or official pretrained model. Results for AIDE [32] are cited from its original paper. The remaining methods are cited from DRCT [4].

	Components		**Different Generators**						**Avg.**
Method	**TTSD**	**SS**	**3DGAN**	**LDM**	**LSGM**	**StyleGAN2**	**StyleGAN3**	**Transformers**	**Acc.(%)**
Baseline	✖	✖	59.85	67.70	70.80	65.60	66.95	67.11	66.34
JointCLIP	**N.A**	**N.A**	71.95	79.25	88.04	69.90	73.40	81.13	77.28
JointTrain	✖	✖	58.10	74.25	75.13	72.10	72.95	68.28	70.13
Ours-TTSD	✔	✖	58.40	95.20	96.18	70.95	81.30	95.41	82.91
Ours-ATTSD	✔	✔	**63.70**	**95.25**	**96.59**	**73.50**	**82.25**	**94.59**	**84.31**

Table 4. Ablation results on the DF3 dataset. TTSD and SS abbreviate Test-Time Semantic Debiasing and Semantic Suppression strategy, respectively. JointCLIP combines features before the last FC layer with CLIP visual embeddings via concatenation. JointTrain refers to regular testing of the variant jointly trained with the semantic alignment task, without the test-time debiasing phase.

families is significantly more challenging than generalizing within the same family. For example, methods such as NoDown, UniFD, and AIDE achieve nearly 90% accuracy on all GAN-based subsets but experience a substantial drop in performance on DM-based fakes. This indicates that only when it comes to generalizing across generator families does the generator factor become a primary constraint. Our method, by directly adapting feature learning to the test distribution, consistently outperforms existing approaches for both GAN-based and DM-based fakes. Among existing methods, NPR stands out by identifying shared up-sampling artifacts common to both GAN and DM families, making it less sensitive to generator variations. Nevertheless, our method still outperforms NPR by 8.8% in accuracy.

Generalization from DM-based generators. Following the setting of GenImage, we train the model on the SDv1.4-based training set and evaluate it on test subsets of various generative models. The comparative results with existing methods are presented in Tab. 3. The GenImage test set contains 7 DM subsets and 1 GAN subset. Although the training generator shifts to DM, a similar limitation persists. Almost all previous methods suffer significant performance drops on the BigGAN subset, as it originates from a generator family different from the training set. DRCT, the latest state-of-the-art method, demonstrates relatively strong generalizability by generating reconstructed images to capture generative artifacts. Rather than generative factors, our method places greater emphasis on addressing limitations posed by semantic factors, making our generalization less sensitive to generator shifts during testing.

Overall, the performance on both datasets underscores the exceptional generalizability of our method in cross-generator detection. These results demonstrate that our generalizability is not confined to cross-semantic scenarios but also encompasses generator variations.

5.4. Ablation study

As aforementioned, we use the DF3 dataset—composed entirely of face-centered images—as the validation set to finalize test settings and assess the contributions of each component. The variants in following experiments are trained on UniForen's training set and evaluated on different generators of DF3, enabling a simultaneous examination of cross-generator and cross-semantic generalization.

Effect of each component. We denote the detector with the same backbone as "Baseline", which consists solely of the main branch of our architecture without the auxiliary task. Another variant jointly trained with the auxiliary task but excluding the test-time optimization process is denoted as "JointTrain". To demonstrate that the benefits of our ap-

proach are not solely derived from the inclusion of CLIP's semantic space, we introduce a variant referred to as "JointCLIP". This variant concatenates CLIP visual features with the features before the final FC layer for classification.

As shown in Tab. 4, the "Baseline" detector achieves only 66.34% average accuracy. Concatenating CLIP visual features ("JointCLIP") into the model yields an improvement, confirming the effectiveness of CLIP's semantic representations. However, its performance still lags significantly behind our approach. The "JointTrain" variant also shows performance gains. Nevertheless, without test-time optimization, the improvement remains limited. The TTSD variant, without the Semantic-Suppression (SS) strategy, improves baseline performance by 16.57%, highlighting the impact of test-time semantic debiasing. Our final ATTSD version, which integrates the SS strategy, further enhances accuracy to 84.31%. These results collectively emphasize the critical role of each proposed component in effective generalization.

Effect of the Semantic Adaptor. We present the empirical results of finalizing the design of the Semantic Adaptor (SA) through two variants. The first variant, denoted as "Share", allows the SA to share its entire structure and weights with the forgery classifier. It is supplemented only by an additional projection layer to map the output features to the CLIP space. The second variant, denoted as "Frozen", retains the same architecture as our implementation but keeps the SA frozen during the test phase.

As shown in Tab. 5, the "Share" variant exhibits inferior performance. We attribute this to the tightly coupled structure between the two tasks, where the forgery classifier is updated during testing by the semantics-related task, causing it to compromise its ability to maintain a robust forgery classification boundary. Similarly, keeping the SA frozen underperforms, as it forces the feature extractor to compensate excessively for the auxiliary task's objectives, leading to suboptimal adjustments. Our finalized implementation adopts a Y-branch architecture, where the SA is loosely coupled with the forgery classifier and remains finetuned during the test phase. This design achieves the best performance by serving two key merits: decoupling the tasks to minimize interference and maintaining the flexibility necessary for effective adaptation.

Effect of gradient steps. In real-world detection scenarios, it is impractical to predefine the optimal number of gradient steps for evolving unknown data. Therefore, we analyze the effect of gradient steps on the DF3 dataset to determine the best configuration and ensure consistency across various test datasets. As shown in Fig. 5, detection accuracy across different subsets improves as the step increases. The overall detection accuracy reaches a saturation point after 6 steps. We thus finalize 6 gradient steps for ATTSD-based AGI detection, striking a balance between computational efficiency and detection performance.

SA	3D-GAN	LDM	LSGM	StyleGAN2	StyleGAN3	Transformers	Avg. Acc.(%)
Share	55.05	75.90	79.14	81.25	80.10	78.44	74.98
Frozen	56.75	74.75	63.30	68.30	63.45	93.01	69.93
Ours	**63.70**	**95.25**	**96.59**	**73.50**	**82.25**	**94.59**	**84.31**

Table 5. Effect of Semantic Adaptor (SA) design. "Share" denotes a tightly coupled adaptor sharing weights with the forgery classifier, "Frozen" refers to a fixed adaptor during the test phase, and "Ours" represents the finalized Y-branch architecture with a loosely coupled and finetuned adaptor.

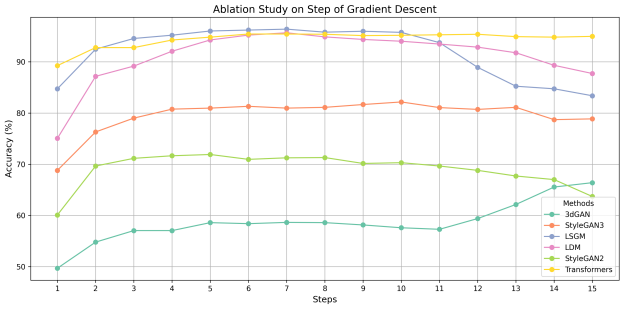

Figure 5. Effect of gradient steps. We present accuracy trends on the DF3 dataset as the number of gradient descent steps increases.

6. Discussion

Limitation and future work. During testing, our method takes 2 $\times$ number_of_steps longer than standard inference, which performs only a single forward pass per sample. As the first work addressing semantic bias for AGI detection, this paper is not as concerned about computational efficiency as improving generalizability. Future work could explore enhancing computational efficiency by designing models that are explicitly optimized during training for faster updates during test time.

Conclusion. This paper introduces *Adaptive Test-Time Semantic Debiasing* (ATTSD), a new method for AGI detection that dynamically aligns the detector's feature space with evolving semantic distributions during testing. By leveraging CLIP's semantic visual space and a semantic-suppression strategy, our method effectively mitigates semantic bias and enhances generalization across diverse semantic and generative model variations. Extensive evaluations on CSAIID, UniForen, and GenImage benchmarks demonstrate that our approach outperforms existing methods, achieving superior cross-semantic and cross-generator generalization. While ATTSD involves additional computational overhead, the significant gains in detection accuracy validate its effectiveness, offering a promising direction for addressing real-world AGI detection challenges.

References

[1] Anjuman Ara, Md Sajadul Alam, et al. A comparative review of ai-generated image detection across social media platforms. *Global Mainstream Journal of Innovation, Engineering & Emerging Technology*, 3(01):11–22, 2024. 1

[2] Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale gan training for high fidelity natural image synthesis. *arXiv*, 2018. 1

[3] Lucy Chai, David Bau, Ser-Nam Lim, and Phillip Isola. What makes fake images detectable? understanding properties that generalize. In *ECCV*, pages 103–120. Springer, 2020. 6

[4] Baoying Chen, Jishen Zeng, Jianquan Yang, and Rui Yang. Drct: Diffusion reconstruction contrastive training towards universal detection of diffusion generated images. In *Forty-first International Conference on Machine Learning*. 2, 6, 7

[5] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009. 5

[6] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in Neural Information Processing Systems*, 34:8780–8794, 2021. 1

[7] Chun-Mei Feng, Kai Yu, Yong Liu, Salman Khan, and Wangmeng Zuo. Diverse data augmentation with diffusions for effective test-time prompt tuning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2704–2714, 2023. 3

[8] Yossi Gandelsman, Yu Sun, Xinlei Chen, and Alexei Efros. Test-time training with masked autoencoders. *Advances in Neural Information Processing Systems*, 35:29374–29385, 2022. 3

[9] Diego Gragnaniello, Davide Cozzolino, Francesco Marra, Giovanni Poggi, and Luisa Verdoliva. Are gan generated images easy to detect? a critical analysis of the state-of-the-art. In *2021 IEEE international conference on multimedia and expo (ICME)*, pages 1–6. IEEE, 2021. 6

[10] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *CVPR*, 2016. 5

[11] Dan Hendrycks, Norman Mu, Ekin Dogus Cubuk, Barret Zoph, Justin Gilmer, and Balaji Lakshminarayanan. Augmix: A simple method to improve robustness and uncertainty under data shift. In *International conference on learning representations*, page 5, 2020. 3

[12] Yan Ju, Shan Jia, Jialing Cai, Haiying Guan, and Siwei Lyu. Glff: Global and local feature fusion for ai-synthesized image detection. *IEEE Transactions on Multimedia*, 2023. 5, 6

[13] 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. 1

[14] 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. 5

[15] Zhengzhe Liu, Xiaojuan Qi, and Philip HS Torr. Global texture enhancement for fake face detection in the wild. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8060–8069, 2020. 7

[16] 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. 5

[17] Lakshmanan Nataraj, Tajuddin Manhar Mohammed, Shivkumar Chandrasekaran, Arjuna Flenner, Jawadul H Bappy, Amit K Roy-Chowdhury, and BS Manjunath. Detecting gan generated fake images using co-occurrence matrices. *arXiv preprint arXiv:1903.06836*, 2019. 6

[18] 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*, pages 24480–24489, 2023. 1, 2, 5, 6, 7

[19] Alan V Oppenheim and Jae S Lim. The importance of phase in signals. *Proceedings of the IEEE*, 69(5):529–541, 1981. 4

[20] Leon N Piotrowski and Fergus W Campbell. A demonstration of the visual importance and flexibility of spatial-frequency amplitude and phase. *Perception*, 11(3):337–346, 1982. 4

[21] Yuyang Qian, Guojun Yin, Lu Sheng, Zixuan Chen, and Jing Shao. Thinking in frequency: Face forgery detection by mining frequency-aware clues. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XII*, pages 86–103. Springer, 2020. 7

[22] 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. 2

[23] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10684–10695, 2022. 1, 5

[24] Zeyang Sha, Zheng Li, Ning Yu, and Yang Zhang. De-fake: Detection and attribution of fake images generated by text-to-image generation models. In *Proceedings of the 2023 ACM SIGSAC Conference on Computer and Communications Security*, pages 3418–3432, 2023. 2, 7

[25] Manli Shu, Weili Nie, De-An Huang, Zhiding Yu, Tom Goldstein, Anima Anandkumar, and Chaowei Xiao. Test-time prompt tuning for zero-shot generalization in vision-language models. *Advances in Neural Information Processing Systems*, 35:14274–14289, 2022. 3

[26] Yu Sun, Xiaolong Wang, Zhuang Liu, John Miller, Alexei Efros, and Moritz Hardt. Test-time training with self-supervision for generalization under distribution shifts. In *ICML*, pages 9229–9248. PMLR, 2020. 2, 3, 5

[27] 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. 5

[28] 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. 1, 2, 6, 7

[29] Sheng-Yu Wang, Oliver Wang, Richard Zhang, Andrew Owens, and Alexei A Efros. Cnn-generated images are surprisingly easy to spot... for now. In *CVPR*, pages 8695–8704, 2020. 1, 2, 5, 6, 7

[30] Zhendong Wang, Jianmin Bao, Wengang Zhou, Weilun Wang, Hezhen Hu, Hong Chen, and Houqiang Li. Dire for diffusion-generated image detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 22445–22455, 2023. 2, 6, 7

[31] Qinwei Xu, Ruipeng Zhang, Ya Zhang, Yanfeng Wang, and Qi Tian. A fourier-based framework for domain generalization. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 14383–14392, 2021. 4

[32] 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. 1, 2, 6, 7

[33] Cai Yu, Peng Chen, Jiao Dai, Xi Wang, Weibo Zhang, Jin Liu, and Jizhong Han. Focus by prior: Deepfake detection based on prior-attention. In *2022 IEEE International Conference on Multimedia and Expo (ICME)*, pages 1–6. IEEE, 2022. 2

[34] Fisher Yu, Ari Seff, Yinda Zhang, Shuran Song, Thomas Funkhouser, and Jianxiong Xiao. Lsun: Construction of a large-scale image dataset using deep learning with humans in the loop. *arXiv preprint arXiv:1506.03365*, 2015. 5

[35] Jiahui Yu, Yuanzhong Xu, Jing Yu Koh, Thang Luong, Gunjan Baid, Zirui Wang, Vijay Vasudevan, Alexander Ku, et al. Scaling autoregressive models for content-rich text-to-image generation. *arXiv preprint arXiv:2206.10789*, 2(3):5, 2022. 5

[36] 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. 6

[37] Haonan Zhong, Jiamin Chang, Ziyue Yang, Tingmin Wu, Pathum Chamikara Mahawaga Arachchige, Chehara Pathmabandu, and Minhui Xue. Copyright protection and accountability of generative ai: Attack, watermarking and attribution. In *Companion Proceedings of the ACM Web Conference 2023*, pages 94–98, 2023. 1

[38] 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. *Advances in Neural Information Processing Systems*, 36, 2024. 2, 5