Automated In-the-Wild Data Collection for Continual AI Generated Image Detection

Thanasis Pantsios
CERTH-ITI
Thessaloniki, n/a, Greece
apantsios@iti.gr

Dimitrios Karageorgiou
CERTH-ITI
Thessaloniki, n/a, Greece
dkarageo@iti.gr

Christos Koutlis
CERTH-ITI
Thessaloniki, n/a, Greece
ckoutlis@iti.gr

George Karantaidis
CERTH-ITI
Thessaloniki, n/a, Greece
karantai@iti.gr

Olga Papadopoulou
CERTH-ITI
Thessaloniki, n/a, Greece
olgapapa@iti.gr

Symeon Papadopoulos
CERTH-ITI
Thessaloniki, n/a, Greece
papadop@iti.gr

Abstract

The rapid advancement of generative Artificial Intelligence (AI) has introduced significant challenges for reliable AI-generated image detection. Existing detectors often suffer from performance degradation under distribution shifts and when encountering newly emerging generative models. In this work, we propose a data-centric continual adaptation framework for updating detectors in evolving environments. We show that both in-the-wild data and generator-driven data are essential for adapting detectors. We introduce an automated, weakly supervised pipeline for constructing in-the-wild datasets through fact-check article retrieval. Additionally, we demonstrate that incorporating even a small amount of generator-driven data during training enables effective adaptation to newly emerging models, while combining it with in-the-wild data within a continual learning framework enables robust adaptation and mitigates catastrophic forgetting. Extensive experiments on two state-of-the-art detectors show significant improvements of +9.14% and +8% in average accuracy, respectively. The proposed dataset and model checkpoints are publicly available at https://mever-team.github.io/WildFC/.

CCS Concepts

• **Information systems** → **Information extraction**; • **Computing methodologies** → **Lifelong machine learning**.

Keywords

AI Generated Image Detection, Continual Learning, Automated Data Collection, Information Extraction

ACM Reference Format:
Thanasis Pantsios, Dimitrios Karageorgiou, Christos Koutlis, George Karantaidis, Olga Papadopoulou, and Symeon Papadopoulos. 2026. Automated In-the-Wild Data Collection for Continual AI Generated Image Detection. In *The 5th ACM International Workshop on Multimedia AI against Disinformation (MAD '26), June 16–19, 2026, Amsterdam, Netherlands.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3810988.3812662

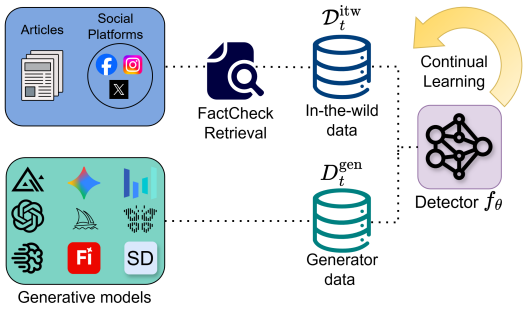

Figure 1: The proposed framework combines regular data collection from in-the-wild data $\mathcal{D}_t^{\text{itw}}$ via fact-check article retrieval and generator-driven data $\mathcal{D}_t^{\text{gen}}$ within a continual learning pipeline to adapt detectors f_θ.

1 Introduction

Generative Artificial Intelligence (AI) is rapidly evolving and playing an important role across many sectors [38]. Advances in generation quality have led to photorealistic images that humans often cannot reliably distinguish from real ones. Capabilities have also expanded beyond text-to-image generation to include editing, style transfer, and multimedia creation [6, 45]. User-friendly interfaces, open models and reduced costs have led to a massive increase in generated content, which increases the risk of misuse for disinformation, fraud, non-consensual content, and political manipulation calling for reliable and content verification mechanisms [38, 45].

In this context, AI-generated image detection (AID) has emerged as a critical challenge. Various approaches have been proposed, from early CNN-based detectors for GAN-generated images [42] to recent approaches targeting diffusion models, including spatial-domain methods [39], spectral analysis [16], and Vision-Language Models (VLMs) [8, 18]. These perform well on controlled benchmarks, but struggle to generalize to out-of-distribution samples,

This work is licensed under a Creative Commons Attribution 4.0 International License.

MAD '26, Amsterdam, Netherlands

© 2026 Copyright held by the owner/author(s).

ACM ISBN 979-8-4007-2700-9/26/06
https://doi.org/10.1145/3810988.3812662

particularly those from unseen generative models, post-processing operations, and real-world data [17, 44] leading to large performance drops in practical settings [19].

As generative models evolve with new capabilities and user interactions, the distribution of AI-generated images is constantly changing, requiring continuous adaptation of detection models. Recent approaches address this challenge through continual and online learning frameworks [1, 9, 10, 24, 43], primarily focusing on adapting detectors to new generators. However, many models are closed-source and computationally expensive, limiting access to representative training data. Moreover, controlled pipelines fail to capture post-processing, editing, and platform-specific transformations in real-world content.

To this end, we propose a continual data collection and learning framework for adapting detectors under distribution shift, as depicted in Fig. 1. Our framework jointly leverages generator-driven and in-the-wild data, allowing detectors to adapt to both newly emerging models and real-world variations. While detector architectures improve, we argue that a data-centric perspective provides a necessary direction to address the evolving nature of AID. Our approach introduces a fact-check retrieval pipeline that enables automated, weakly supervised dataset construction by incorporating in-the-wild data from real-world sources. We show that even small amounts of such data significantly improve performance. We evaluate our framework on two state-of-the-art detectors, RINE [18] and SPAI [16], which remain vulnerable to distribution shifts and emerging generative models and represent diverse approaches, including spectral and CLIP-based methods, making them suitable for assessing robust generalization. In summary:

- We propose a continual data collection and learning framework that adapts detectors to evolving data distributions and generalizes across detector architectures.
- We introduce a fact-check retrieval pipeline for automated, weakly supervised creation of in-the-wild datasets for AID.
- We introduce an evolving dataset of AI-generated images, including 2,884 in-the-wild instances and 5,439 images generated by 19 recent generative models. In addition, we present a large dataset of 213,674 real images collected in the wild.
- We demonstrate the general applicability of our framework across two different detection models, consistently improving state-of-the-art performance by +9.14% for SPAI and +8% for RINE in average accuracy.

2 Related Work

This section reviews advances in AID, including methods, datasets, and continual adaptation strategies. We highlight limitations in handling real-world data and evolving models, motivating continuous and adaptive frameworks.

2.1 AI-Generated Image Detection

Early AID works focus primarily on GAN-based generators. Wang et al. [42] demonstrate that detectors trained on a limited set of generators can generalize to unseen ones, indicating shared generative artifacts. Subsequent works explored more robust detection strategies. SAFE [20] adopts a lightweight framework for both GAN- and diffusion-generated images, incorporating simple transformations and a local-awareness training strategy. CLIP-based approaches [8] show strong generalization using penultimate-layer features and a simple SVM classifier. RINE [18] explores CLIP's representation capabilities by leveraging intermediate Transformer blocks that capture low-level visual artifacts. It employs a linear mapping to project these features into a forgery-aware space and a trainable importance estimator to weight each block's contribution. SPAI [16] models the spectral distribution of real images using masked spectral learning and treats generated images as out-of-distribution samples through a spectral reconstruction module.

While detectors excel on controlled benchmarks, several works show significant degradation on realistic scenarios and unseen generators [15]. Yan et al. [44] posit that detectors often misclassify challenging AI-generated images, indicating that the problem is far from solved. Similarly, Konstantinidou et al. [17] show that models trained on benchmark data struggle with real-world variations, and that incorporating in-the-wild data during training substantially improves performance. To address this challenge, B-Free [14] introduces a bias-free approach that constructs semantically aligned real–fake pairs via controlled generation, isolating synthesis artifacts and achieving strong generalization. While this improves data quality in a static setting, our work follows an orthogonal direction by addressing the temporal evolution of data distributions, modeling AID as a non-stationary problem and enabling continuous adaptation through in-the-wild and generator-driven data.

2.2 Datasets and Benchmarks for AID

Recent advances in the area highlight the importance of dataset design, including curated benchmarks, in-the-wild collections, and realistic evaluation protocols. Twigma [5] introduces a real-world dataset collected from Twitter, demonstrating their characteristics and temporal evolution. Corvi et al. [7] introduce a dataset including diffusion-generated images to extend AID beyond GAN images, highlighting that different generative models produce distinct artifacts and showing performance degradation under realistic distortions. RealHD [48] constructs a high-quality and diverse dataset for AID, addressing limitations of earlier benchmarks in image quality, prompt complexity and diversity. RealHD does not rely solely on text-to-image pipelines but includes multiple generation settings such as inpainting, refinement, and face swapping, with images generated by state-of-the-art diffusion models. Building on these directions, we propose a unified update framework that integrates both in-the-wild and generator-driven data, moving beyond static datasets toward a continuously evolving data distribution.

While not specifically designed for AID, recent works explore structured and synthetic data generation for well-curated datasets. OpenGPT-4o-Image [6] introduces a comprehensive dataset with a hierarchical task taxonomy and automated data generation using GPT-4o, emphasizing structured and diverse data construction. Similarly, Echo-4o-Image [46] is a synthetic dataset generated by GPT-4o, providing clean and controllable supervision while covering rare and long-tail scenarios. In addition, RealBench [47] is a benchmark for evaluating the photorealism of generated images. The work introduces a photorealistic text-to-image framework that employs an adversarial approach through a detector reward mechanism, highlighting the evolving nature of the generation process.

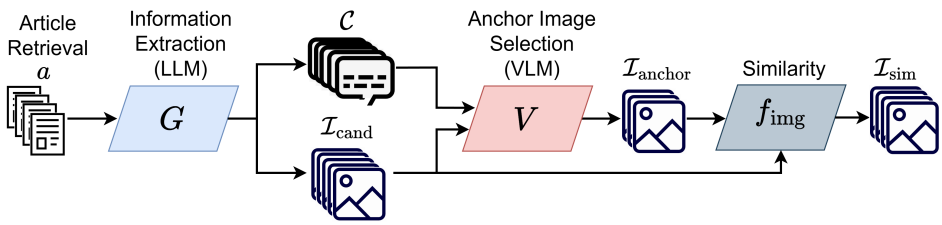

Figure 2: Fact-check retrieval pipeline: Given an article a, an LLM $\mathcal{G}$ extracts textual descriptions C referring to AI-generated images mentioned in a, along with a set of candidate images $\mathcal{I}_{\text{cand}}$. A VLM $\mathcal{V}$ identifies anchor images $\mathcal{I}_{\text{anchor}}$ that are semantically aligned with C, and a similarity function based on an image encoder f_{img} further expands this set to $\mathcal{I}_{\text{sim}}$.

These directions are important for advancing dataset design for AID, improving data construction and curation procedures.

A growing line of work focuses on in-the-wild benchmarks to reflect realistic AID conditions, highlighting the gap between controlled benchmarks and practical deployment [17, 21, 30, 44]. AI-GenBench [32] introduces a temporal benchmark with a chronologically organized dataset, combining multiple existing real and synthetic datasets. In this setting, detectors are trained on earlier generative models and evaluated on newer ones, emphasizing the evolving nature of the problem. RRDataset [19] is introduced as a benchmark designed to evaluate AID under realistic conditions, incorporating diverse scenarios, social media transmission effects, and re-digitization processes, and revealing significant limitations of existing detectors. Our work addresses these challenges by incorporating real-world data through a continuously evolving dataset, while enabling adaptation to emerging generative models through a continual update framework for detectors.

2.3 Continual and Online Adaptation in AID

Several works explore AID under continual learning. S-Prompts [43] propose a prompt-based method where task-specific prompts are learned for sequential domains while keeping the backbone model frozen. CLOFAI [9] introduces a domain-incremental AID benchmark where data are organized into sequential tasks reflecting the evolution of generative models. However, both approaches assume discrete task boundaries, where each incremental step is tied to a specific generative model, which does not reflect the evolving nature of real-world scenarios.

Recent works explore adaptive AID frameworks. Epstein et al. [10] propose an online detection framework, providing insights into temporal generalization through a streaming setup where detectors are trained on a set of generators and evaluated on future unseen ones. Similarly, E3 [1] presents an adaptive approach that addresses newly emerging generators with limited data by learning generator-specific expert embedders and combining them through a fusion mechanism. LiteUpdate [24] introduces a lightweight strategy that updates detectors using small task-specific modules, along with a representative sample selection mechanism. It also incorporates a model merging scheme to balance adaptation to new generators while mitigating catastrophic forgetting. Pellegrini et al. [33] offer a comprehensive analysis of architecture-agnostic design choices for detectors, including augmentation, preprocessing, multiclass supervision, and incremental learning. For continual adaptation, they explore replay-based methods such as harmonic and class-balanced replay across successive generator windows. However, these approaches remain generator-centric and rely on updates tied to individual generative models, emphasizing architectural adaptation without explicitly incorporating in-the-wild data.

3 Methodology

3.1 Problem Formulation

AID is a binary classification problem defined over an input space $\mathcal{X}$ (images) and label space $\mathcal{Y} = \{0, 1\}$, where $y = 1$ denotes AI-generated images and $y = 0$ real images. The data is generated according to a joint distribution $P_t(x, y)$ that evolves over time t. We consider a non-stationary environment, where the underlying data distribution changes sequentially over time:

$$P_{t_0}(x, y) \neq P_t(x, y), \quad \text{for } t > t_0. \tag{1}$$

This temporal evolution violates the i.i.d. assumption and leads to dataset shift [25], making AID a rapidly evolving problem. The first source of distribution shift arises from the continuous development of new generative models, architectures, and synthesis pipelines. These introduce new patterns and artifacts, modifying their distribution over time. In addition, the distribution shift arises from changes in user behavior and real-world usage over time. As generative models are adopted in practice, users interact with them in diverse ways (e.g., editing, compositional generation, platform-specific processing), which affects how synthetic content is produced and perceived. Formally, this corresponds to covariate shift:

$$P_t(x \mid y) \neq P_{t_0}(x \mid y), \quad P_t(y \mid x) = P_{t_0}(y \mid x). \tag{2}$$

These two complementary mechanisms motivate the need for multiple data sources to track the evolving distribution. In particular,

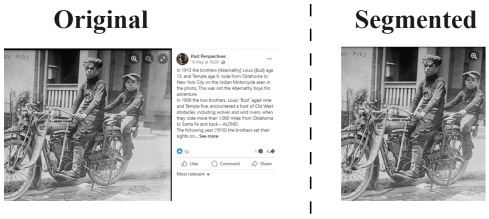

Figure 3: Image segmentation: original (left) and segmented (right) image.

our framework leverages generator-driven data to capture covariate shift induced by evolving generative models, and in-the-wild data to capture covariate shift from real-world usage dynamics. This dual-source strategy enables a better approximation of the distribution $P_t(x, y)$ and supports effective updating of detectors.

3.2 Fact-Check Retrieval

Our framework constructs a dataset of AI-generated images by leveraging fact-check articles, as depicted in Fig. 2. Fact-check articles are treated as sources of weak supervision describing AI-generated images.

An information extraction process is performed using a large language model $\mathcal{G}$, which analyzes an article a and extracts textual descriptions corresponding to AI-generated images mentioned in the article narrative. Articles that do not refer to AI-generated images are discarded. Formally, this process is defined as

$$C = \mathcal{G}(a, p_1) = \{c_i\}_{i=1}^{K} \tag{3}$$

where p_1 is a task-specific instruction prompt, as depicted in Fig. 5, and $\mathcal{G}$ represents the LLM-based extraction function.

The set C contains the textual descriptions derived from article a. Each description c_i corresponds to the visual content of the i-th AI-generated image referenced in the article, while K denotes the number of extracted descriptions.

For each article a, we collect a set of candidate images $\mathcal{I}_{\text{cand}} = \{x_j\}_{j=1}^{N}$ associated with the article, where x_j denotes the j-th candidate image and N represents the number of collected images. The goal of this process is to automatically determine which of the candidate images correspond to the AI-generated content described in the article and which are unrelated and should be discarded.

To this end, an anchor image selection process is performed to identify images that correspond to the extracted textual descriptions. To achieve this, we employ a VLM that evaluates the semantic compatibility between candidate images and textual descriptions.

Given a candidate image x_j and a textual description c_i, the VLM produces a similarity score measuring the alignment between the image and the description. Since an article may describe multiple AI-generated images, the final similarity score for candidate image x_j is defined as the maximum similarity across all textual descriptions:

$$s_j = \max_{c_i \in C(a)} \mathcal{V}(x_j, c_i, p_2) \tag{4}$$

where $\mathcal{V}$ denotes the VLM scoring function and p_2 is the task-specific prompt, as detailed in Fig. 6.

Candidate images whose similarity scores exceed a predefined anchor selection threshold τ_{anchor} are considered anchor images:

$$\mathcal{I}_{\text{anchor}} = \{x_j \in \mathcal{I}_{\text{cand}} \mid s_j \geq \tau_{\text{anchor}}\} \tag{5}$$

To further expand the anchor set, we compute the visual similarity between anchor and candidate images using an image encoder f_{img}:

$$s(x_a, x_j) = \text{sim}\big(f_{\text{img}}(x_a), f_{\text{img}}(x_j)\big), \tag{6}$$

where $x_a \in \mathcal{I}_{\text{anchor}}$ and $x_j \in \mathcal{I}_{\text{cand}}$.

Candidate images are selected if their similarity with at least one anchor image exceeds a predefined threshold τ_{sim}:

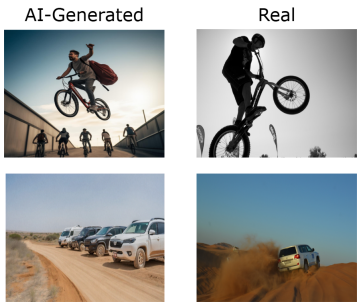

Figure 4: Examples of semantically aligned real–fake pairs.

$$\mathcal{I}_{\text{sim}} = \{x_j \in \mathcal{I}_{\text{cand}} \mid \max_{x_a \in \mathcal{I}_{\text{anchor}}} s(x_a, x_j) \geq \tau_{\text{sim}}\} \tag{7}$$

The final image set associated with article a is defined as

$$\mathcal{I}_{\text{final}} = \mathcal{I}_{\text{anchor}} \cup \mathcal{I}_{\text{sim}} \tag{8}$$

Additionally, to isolate candidate visual regions corresponding to the AI-generated content from complex images (e.g., social media screenshots), we apply an image segmentation process, producing segmented regions. An example is shown in Fig. 3. The segmented dataset is therefore defined as

$$\mathcal{I}_{\text{seg}} = \bigcup_{x \in \mathcal{I}_{\text{final}}} \mathcal{S}(x) \tag{9}$$

where $\mathcal{S}(x)$ denotes the set of visual segments.

Finally, to construct semantically aligned real–fake image pairs, we retrieve real images from a pool of real images $\mathcal{R}$ that are visually similar to the AI-generated images in the final set. Given an AI-generated image $x_f \in \mathcal{I}_{\text{seg}}$ and a real image $r_k \in \mathcal{R}$, we compute image similarity using the same image encoder:

$$s(x_f, r_k) = \text{sim}\big(f_{\text{img}}(x_f), f_{\text{img}}(r_k)\big) \tag{10}$$

The TopK retrieval is performed without replacement to ensure that each real image is selected at most once for a given AI-generated image. The final dataset consists of AI-generated images paired with semantically similar real images:

$$\mathcal{R}_{\text{match}}(x_f) = \text{TopK}_{r_k \in \mathcal{R}}\, s(x_f, r_k) \tag{11}$$

3.3 Continual Adaptive Framework

We propose a general continual data collection and learning framework for adapting detectors under distribution shift, as illustrated in Fig. 1. At regular update intervals, the framework incorporates three

LLM Prompt P_1

```
You are an assistant that analyzes fact-check articles written in {language}.
Your task is to extract information about AI-generated images from the article.
Follow these steps:
  1. Determine whether the article discusses or confirms the use or presence of AI-generated images.
  2. If yes, write a descriptive caption that explains the visual content of the AI-generated image(s), based on how the article describes them.
     - Include specific visual details such as people, objects, actions, clothing, background elements, settings in the image.
     - The caption should be detailed enough to allow comparison between the description and the actual image.
     - Do NOT mention AI-generated in the caption.
  3. Extract and return only URL that directly lead to a page displaying the AI-generated images.
     - Include social media posts (e.g., Twitter, Facebook, Instagram) or webpages that show the AI-generated images.
     - Do NOT include URLs that point to videos, profile pages, home pages or pages that merely mention the image without displaying it
Return a JSON object with:
   - ai_image: True or False
   - image_captions: a list of short captions that describes the AI-generated images
   - image_urls: a list of valid URLs that relate to AI-generated Image as defined above
Respond in pure JSON. Do not include explanations or extra text.
```

Figure 5: Instruction prompt p_1 template for LLM $\mathcal{G}$.

VLM Prompt P_2

```
You are given the following inputs: A list of image captions written in {language}, an image. Your main goal is to assign an
image_match_score from 0 to 10, based on how well the visual content of the image matches any of the captions.
Compare the image with each caption separately and base your score on the strongest match.
Return a JSON object with: image_match_score: an integer from 0 to 10.
Scoring Guide:
  -  10: Exact match — the image shows exactly what the caption describes.
  -  7–9: Strong match — most key elements are clearly present.
  -  4–6: Partial match — some elements match, but others are missing, unclear, or mixed with unrelated content.
  -  1–3: Weak match — only minimal or vague similarity
  -      0: No match — the image is completely unrelated or irrelevant.
Rules:
  - Focus only on visible visual content (e.g., people, objects, settings, actions). Ignore overlays, metadata, watermarks,
    or surrounding UI unless integral to the meaning.
  - If the image is a social media post or a platform UI screenshot, score it based on that image's content.
  - If the image contains multiple visuals (e.g., a collage or grid of images), and only one part matches a caption while
    others are unrelated, assign a score no higher than 6, depending on how prominent and clear the relevant part is.
Output Format: { image_match_score: <integer from 0 to 10> }
```

Figure 6: Instruction prompt p_2 template for VLM $\mathcal{V}$.

sources: newly collected in-the-wild data, samples from recently released generative models, and a continual learning mechanism.

At each update round t, we collect two complementary datasets. First, we gather in-the-wild data $\mathcal{D}_t^{\text{itw}}$ (e.g., WildFC; see Sec. 4.1), which reflects real-world image distributions and evolving artifact patterns. Second, we construct a dataset of recent generators $\mathcal{D}_t^{\text{gen}}$ (e.g., AIGenImages2026; see Sec. 4.2). To mitigate catastrophic forgetting, we maintain a replay buffer $\mathcal{M}_{t-1}$, constructed by sampling a fixed proportion ρ of the accumulated data observed up to round t. The training set at round t is defined as:

$$\mathcal{D}_t = \mathcal{D}_t^{\text{itw}} \cup \mathcal{D}_t^{\text{gen}} \cup \mathcal{M}_{t-1}. \tag{12}$$

Given the aggregated dataset $\mathcal{D}_t$, the detector is updated by minimizing a loss function specific to the detector architecture:

$$\theta_t = \arg\min_{\theta} \ \mathcal{L}(f_\theta, \mathcal{D}_t). \tag{13}$$

This framework enables effective continuous adaptation. In-the-wild data enhances robustness to real-world distribution shifts, while generator-specific data facilitates rapid adaptation to emerging models. A continual learning component mitigates catastrophic forgetting, ensuring stable performance. While we instantiate this framework with a specific data collection pipeline and a replay-based strategy, its modular design remains compatible with alternative collection and continual learning methods.

4 Data collection

4.1 Fact-Checked Collection Retrieval

We introduce WildFC, an evolving dataset of in-the-wild AI-generated images, currently comprising 2,884 images collected in 2025 through the automated fact-check retrieval pipeline described in Sec. 3.2. In total, 3,841 articles were retrieved from fact-checking databases and resources, including the Google Fact Check Tools [13] and the Database of Known Fakes (DBKF) [27], using queries related to AI-generated content. Among these, 1,539 articles were identified as relevant using the Qwen3-8B-FP8 [40] language model, which was used to detect references to AI-generated images, extract image URLs, and generate corresponding captions.

From the selected articles, a total of 10,387 candidate images were collected using web scraping tools such as Crawl4AI [41] and gallery-dl [11]. For AI-generated image selection, we employed the Qwen2.5-VL-7B-Instruct [34] VLM, which filtered the dataset to 2,884 relevant images. The set consisted mainly of JPEG files at 73.86%, followed by PNG at 22.61%, and WEBP at 3.54%, spanning a resolution range of 0.06 MP to 16.78 MP. Finally, to isolate meaningful visual regions and remove platform-specific overlays or UI elements, we applied image segmentation using Grounding DINO Tiny [23]. This process produced 2,298 segmented image samples, which serve as augmented views of the original content.

A validation set of 300 images, representing 10.40% of the dataset, was examined to assess this weakly supervised automated labeling. The images were manually annotated, and a precision of 91.95% was achieved. Although a small amount of noise was observed, it mainly occurs in more complex cases, such as ambiguous instances where the fact-checker could not provide a definitive verdict, or real images reused out of context to depict new events. The results presented in Sec. 5 show that this noise is limited and that our dataset improves the performance of detectors.

4.2 Data from Recent Generators

We construct AIGenImages2026, a dataset of images generated by 19 recent text-to-image models, as depicted in Table 1. As our framework targets lightweight and cost-effective adaptation, we limit the number of samples to a maximum of 305 images per generator. We reserve 10% of the generated data exclusively for evaluation. Following recent trends in synthetic data generation [6, 46, 47], the prompts are designed to ensure both diversity and realism. We employ three groups of prompts with different perspectives and scopes. First, we generate 105 realistic text-to-image prompts using GPT-5.2 [28], covering 21 real-world categories. Second, we include 100 prompts targeting more complex generation settings, including spatial reasoning, compositional constraints, and stylistic variations based on the OpenGPT-4o [6] dataset. Third, we incorporate 100 prompts derived from the VisualNews [22] dataset by selecting captions from representative topics and refining them using GPT-5.2, where identities and sensitive attributes are neutralized and visual semantics preserved. In total, the dataset contains 5,439 generated images, with 4,880 images used for training and 559 for testing. Most images are generated through the fal.ai API [12], while a small subset is collected manually for models requiring separate access. For certain models, such as Midjourney v7, FLUX.2 [dev], and Adobe Firefly Image 5, fewer samples are available due to access limitations and API restrictions, while all models remain consistently represented across training and evaluation.

Model	Date	Size	Split (Train/Test)
SDXL	2023-11	305	274 / 31
FLUX.1 [pro] v1.1, SD 3.5 Med	2024-10	305	274 / 31
Reve Img 1.0, HiDream I1 Dev	2025-03	305	274 / 31
GPT-Img 1, Ideogram 3	2025-04	305	274 / 31
Midjourney v7	2025-04	301	270 / 31
Imagen 4	2025-05	305	274 / 31
Gemini 2.5 Flash Img	2025-10	305	274 / 31
Firefly Img 5	2025-10	150	133 / 17
FLUX.2, Z Img Turbo	2025-11	305	275 / 31
FLUX.2 [pro]	2025-11	305	274 / 31
FLUX.2 [dev]	2025-11	205	182 / 23
Gemini 3 Pro Img	2025-11	305	276 / 31
FLUX.2 [max]	2025-12	205	182 / 23
GPT-Img 1.5, Seedream 4.5	2025-12	305	274 / 31

Table 1: Overview of the AIGenImages2026 dataset.

Table 2: Comparison of all detectors across benchmark datasets. Each entry reports AUC / ACC (%). For each dataset and each metric, the best result is highlighted in bold, while the second best is underlined.

Method	Recent Generators					ITW Data				
	AIGenImages2026	MNW	Echo-4o	Nano-Consistent	Synthbuster	Chameleon	DeepFake-Eval	ITW-SM	MediaEval	AVG
CNN Detect [42]	45.59 / 49.91	50.60 / 49.91	53.12 / 52.60	40.77 / 48.85	38.30 / 49.09	48.40 / 56.99	48.71 / 39.03	47.90 / 49.99	49.54 / 49.89	46.99 / 49.58
NPR [39]	68.28 / 59.84	66.31 / 60.29	60.43 / 56.60	48.48 / 47.74	47.32 / 53.51	65.35 / 61.33	55.80 / 57.37	56.23 / 54.85	63.59 / 59.34	59.09 / 56.76
SAFE [20]	71.11 / 69.68	47.86 / 49.52	79.44 / 65.60	45.22 / 48.95	59.60 / 52.73	57.16 / 59.14	52.40 / 43.36	49.53 / 49.86	53.40 / 50.24	57.30 / 54.34
LaDeDa [3]	60.62 / 56.08	59.39 / 56.55	79.24 / 72.55	44.58 / 45.30	62.93 / 59.99	65.11 / 61.86	62.54 / 52.11	74.48 / 67.21	69.91 / 63.76	64.31 / 59.49
RINE [18]	75.70 / 60.82	84.41 / 66.98	75.70 / 68.50	**85.58** / 60.07	83.76 / 83.01	39.27 / 44.77	71.84 / 49.45	70.21 / 56.18	64.71 / 54.65	72.35 / 60.49
ITW_RINE [17]	90.80 / 71.74	93.50 / 81.88	50.89 / 48.75	68.81 / 63.81	90.87 / 73.42	88.19 / 79.60	84.56 / 75.56	96.23 / 81.49	89.08 / 76.41	83.66 / 72.52
ITW_SPAI [17]	90.05 / 64.67	92.00 / 75.92	**86.40** / 60.40	74.81 / 55.80	**97.45** / **91.94**	90.14 / 79.09	88.73 / 71.50	**98.13** / **92.81**	94.79 / 86.91	90.28 / 75.45
RINE (ours)	**97.75** / **92.13**	**95.74** / **88.59**	**88.45** / 76.80	78.35 / 71.01	94.81 / 77.75	89.64 / 78.58	87.59 / 78.63	95.74 / 82.50	91.09 / 78.80	91.02 / 80.53
SPAI (ours)	95.93 / 89.09	93.90 / 84.08	86.27 / **79.75**	84.31 / **71.43**	95.54 / 89.86	**92.38** / **84.90**	**90.09** / **82.33**	97.72 / 90.65	**94.83** / **89.20**	**92.33** / **84.59**

4.3 Real Data Collection

Real in-the-wild data is crucial for effectively updating detection models. We collect data from two primary sources. First, real images are gathered from news websites using the NewsAPI [26], covering over 100 official media outlets across multiple categories, including general, science, health, entertainment, technology, business, and sports. Articles related to AI-generated content are explicitly filtered out using query parameters. Second, additional images are collected from social media platforms such as Facebook, Instagram, and X by monitoring 25 official news media accounts. The diversity of platforms and sources is essential for constructing a non-biased training set that reflects real-world data distributions.

In total, we construct a dataset of 213,674 real images collected between August and December 2025, with 155,761 images sourced from news outlets and 57,913 from social media. This continuously evolving dataset serves as a candidate pool from which semantically similar real images are retrieved for each AI-generated sample using a similarity-based matching strategy, forming aligned real–fake training pairs, as depicted in Fig. 4. This process results in a balanced, challenging, and semantically consistent training dataset.

5 Experiments

5.1 Evaluation Datasets and Metrics

We evaluate our approach on datasets covering both recent generators and in-the-wild conditions. For recent generators, we use a held-out test set of 558 images from AIGenImages2026 (see Table 1, MNW [36], an evolving benchmark covering 51 generators, and Synthbuster [2]. Additionally, we include a publicly released subset of 5,000 images from Nano-Consistent [46] and 1,000 images from Echo-4o [46] generated with the GPT-4o model. For in-the-wild data, we use the challenging Chameleon [44], and ITW-SM [17], an in-the-wild dataset of 10,000 real and synthetic images collected from four social media platforms. Additionally, we use MediaEval-ITW [30], which combines real images from datasets with synthetic images augmented through realistic transformations and 1,975 images from Deepfake-Eval [4], a multimodal benchmark based on real-world content. This combination enables evaluation across both controlled generation and realistic deployment scenarios.

We evaluate detection performance using two standard metrics for AID, Accuracy (ACC) and the area under the receiver operating characteristic curve (AUC). AUC measures the overall ranking capability independent of a fixed decision threshold, providing a global view of its discriminative ability. ACC reflects performance at a specific operating point and therefore provides a more practical measure. ACC is calculated using a threshold of 0.5.

5.2 Implementation details

For fact-check retrieval, we use $\tau_{\text{anchor}} = 0.8$, $\tau_{\text{sim}} = 0.75$, TopK = 500, and a segmentation threshold of 0.4, with CLIP ViT-L/14 [35] as the image encoder f_{img}. Compared methods are evaluated using checkpoints from the official repositories, using the SIDBench [37] framework for CNNDetect, NPR, and RINE, and AIGI-Bench [21] for SAFE and LaDeDa trained on WildRF [3]. Additionally, we evaluate the versions of RINE and SPAI proposed in [17], adopting the best-performing reported configuration with a DINOv2 [29] backbone trained on TWIGMA [5] and LDM [7].

Experiments are conducted on the RINE and SPAI models proposed in [17], trained on WildFC, AIGenImages2026, and real images (see Sec. 4), while employing a replay memory ρ of 5%. RINE is trained for 1 epoch with a learning rate of 1×10^{-3} and a batch size of 16 on an NVIDIA RTX 4090 GPU. SPAI is trained for 3 epochs with a learning rate of 2.5×10^{-7}, using a batch size of 24, mixed-precision training, and gradient accumulation over 2 steps implemented in PyTorch [31] on an NVIDIA RTX 5090 GPU.

5.3 Main Analysis

A set of state-of-the-art detectors is evaluated to analyze the impact of distribution shift across recent generator data and in-the-wild data, as summarized in Table 2. Standard detectors such as CNN Detect [42], NPR [39], and SAFE [20] exhibit limited generalization ability, achieving average accuracies of 49.58%, 56.76%, and 54.34%, respectively. While LaDeDa [3] and RINE reach higher averages of 59.49% and 60.49%, performance remains inconsistent across domains. Notably, RINE degrades on in-the-wild datasets, falling to 44.77% on Chameleon and 49.45% on Deepfake-Eval. These results indicate that models trained on static data distributions fail to generalize under evolving conditions.

Methods trained on in-the-wild data, such as ITW_RINE and ITW_SPAI, show improved robustness on real-world data, achieving 81.49% and 92.81% accuracy on ITW-SM, respectively. However, they exhibit degraded performance on Echo-4o and Nano-Consistent with 48.75% and 63.81% accuracy, highlighting the imbalance between robustness to real-world data and generalization to newly emerging generators.

Our framework yields consistent improvements across both domains. For RINE, average accuracy increases from 72.52% to 80.53%,

Table 3: Ablation study of the proposed framework components. Each entry reports AUC / ACC (%).

Training Data		Replay	Recent Generators					ITW Data				
WildFC	AI-Gen		AIGenImages2026	MNW	Echo-4o	Nano-Consistent	Synthbuster	Chameleon	DeepFake-Eval	ITW-SM	MediaEval	AVG
✗	✗	✗	90.80 / 71.74	93.50 / 81.88	50.89 / 48.75	68.81 / 63.81	90.87 / 73.42	88.19 / 79.60	84.56 / 75.56	96.23 / 81.49	89.08 / 76.41	83.66 / 72.52
✓	✗	✗	94.18 / 75.49	93.45 / 77.34	77.93 / 51.75	**84.40** / **74.81**	95.16 / 85.62	84.23 / 74.40	84.03 / 74.31	95.33 / **87.85**	89.47 / 80.13	88.69 / 75.74
✓	✗	5%	94.35 / 78.62	94.79 / 82.97	67.54 / 60.60	79.53 / 72.65	93.64 / 80.30	**90.93** / **81.04**	86.57 / 77.59	**96.81** / 85.14	90.88 / **80.89**	88.34 / 77.76
✗	✓	5%	**97.75** / 91.23	95.85 / 89.22	89.25 / 75.05	72.84 / 66.99	95.08 / 75.55	88.54 / 75.90	86.40 / 77.28	95.61 / 78.98	90.21 / 75.55	90.17 / 78.42
✓	✓	✗	95.74 / 87.30	90.97 / 82.44	**98.73** / **90.10**	76.11 / 70.21	**96.41** / **87.37**	78.86 / 66.24	84.57 / 76.45	90.60 / 80.52	85.21 / 77.03	88.58 / 79.74
✓	✓	3%	97.57 / 91.59	95.18 / 87.12	92.71 / 82.10	76.75 / 70.80	93.49 / 80.97	84.99 / 73.23	87.32 / 78.43	94.90 / 83.53	89.83 / 80.22	90.30 / **80.89**
✓	✓	5%	**97.75** / **92.13**	95.74 / 88.59	88.45 / 76.80	78.35 / 71.01	94.81 / 77.75	89.64 / 78.58	**87.59** / **78.63**	95.74 / 82.50	91.09 / 78.80	91.02 / 80.53
✓	✓	10%	97.42 / 91.06	**95.89** / **89.36**	88.65 / 75.50	79.15 / 71.36	94.76 / 75.51	89.89 / 75.91	87.58 / 78.01	95.80 / 78.65	**91.13** / 76.61	**91.14** / 79.11

an absolute gain of 8.01%. Performance on recent generators improves significantly, where Echo-4o accuracy rises from 48.75% to 76.80% and Nano-Consistent from 63.81% to 71.01%, while maintaining strong performance on in-the-wild datasets. Similar behavior is observed for SPAI, which achieves a 9.14% gain, reaching an average accuracy of 84.59%, with improvements across both recent generators and in-the-wild datasets. These results demonstrate that static training is insufficient under evolving data distributions and highlight the importance of a continual, data-centric training pipeline to maintain robust performance over time. Our framework enables balanced adaptation, improving generalization to recent generators while preserving robustness on in-the-wild data.

5.4 Continual Learning Analysis

We evaluate our framework by simulating a continual learning setting using the RINE model across four incremental tasks, defined at 3-month intervals throughout 2025. Starting from a pretrained model (Task 0), we progressively expand the training set with data accumulated up to March, June, September, and December 2025 for Tasks 1–4, respectively. Chronological ordering is based on WildFC image metadata and AIGenImages2026 release dates (see Table 1). This setup constitutes a simulation, as samples from preview versions of generative models may appear earlier in in-the-wild data. A replay buffer is used during training. The results are presented in Fig. 7 and Fig. 8.

The pretrained RINE model (Task 0) performs poorly across several generators in AIGenImages2026 test set. Task 1, yields a clear improvement in average AUC across all benchmarks, with a gain of +3.49%. On AIGenImages2026, the average accuracy increases by +11.10%. Notably, substantial gains are observed for generators not directly included in training. For example, Gemini 3 Pro improves by +12.90% in terms of accuracy, FLUX.2 by +25.81%, and Imagen 4 by +25.81%, highlighting the strong generalization capability of the framework.

A similar trend continues in Task 2, with average AUC rising another +3.64% over Task 1. We observe significant improvements to new generators, such as GPT-Image 1 with +25.80% in terms of accuracy and Gemini 2.5 Flash with +19.36%. In parallel, in-the-wild performance improves compared to Task 0, including Chameleon with +1.97% in terms of AUC, Deepfake-Eval with +3.30%, and MediaEval-ITW with +1.49%.

Task 3 maintains stable performance with a slight −0.56% decrease in average AUC. While AUC improves by +0.33% on ITW-SM and +0.56% on MediaEval-ITW, a minor degradation of −0.16% occurs on Deepfake-Eval, and a more noticeable drop of −4.90% appears in the Echo-4o dataset. Similarly, AIGenImages2026 shows a small overall accuracy decrease of −0.79%. Despite this, improvements persist for several generators, including Gemini 3 Pro (+13.23%

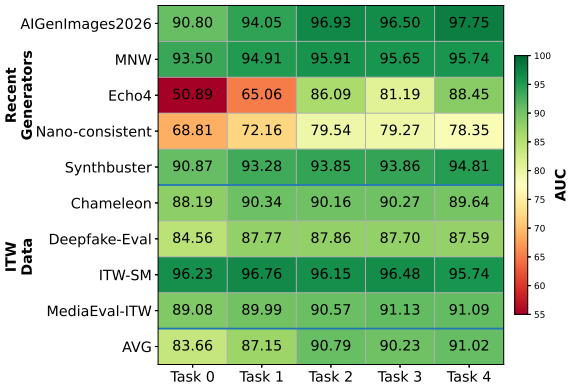

Figure 7: AUC performance under continual learning. Results demonstrate adaptation to generators while maintaining and improving performance on in-the-wild data across tasks.

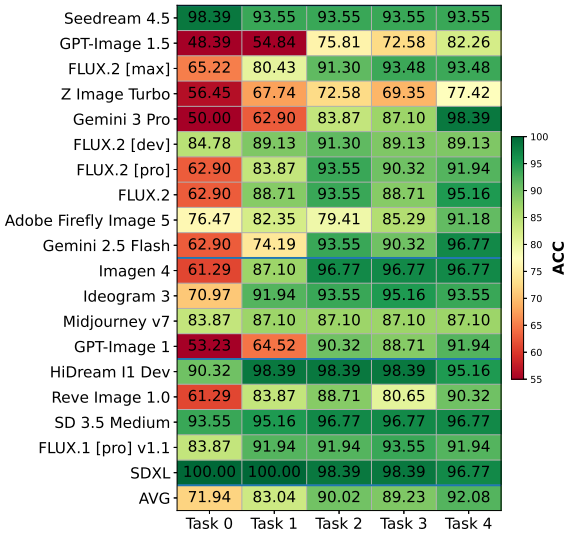

Figure 8: ACC on AIGenImages2026 across generators under continual learning. Results show consistent improvements across tasks, demonstrating effective adaptation while mitigating catastrophic forgetting.

Table 4: Ablation study of the proposed framework across different training data portions. Each entry reports AUC / ACC (%).

Size	Recent Generators					ITW Data				AVG
	AIGenImages2026	MNW	Echo-4o	Nano-Consistent	Synthbuster	Chameleon	DeepFake-Eval	ITW-SM	MediaEval	
0%	90.80 / 71.74	93.50 / 81.88	50.89 / 48.75	68.81 / 63.81	90.87 / 73.42	88.19 / 79.60	84.56 / 75.56	96.23/ 81.49	89.08 / 76.41	83.66 / 72.52
10%	94.07 / 77.82	94.54 / 82.92	62.44 / 58.05	72.49 / 65.42	90.53 / 75.91	89.62 / **81.70**	86.55 / 78.06	**96.50** / **84.06**	89.97 / 78.86	86.30 / 75.87
30%	97.09 / 90.88	**96.16** / **90.19**	81.25 / 72.95	**82.12** / 67.78	94.29 / 74.77	**91.12** / 73.28	87.24 / 77.28	96.17 / 73.33	90.60 / 71.29	90.67 / 76.86
50%	97.49 / 90.52	95.98 / 86.64	77.55 / 72.90	77.30 / 70.91	92.72 / **79.19**	89.08 / 79.16	86.34 / 77.02	95.68 / 83.39	89.90 / **79.44**	89.12 / 79.91
100%	**97.75** / **92.13**	95.74 / 88.59	**88.45** / **76.80**	78.35 / **71.01**	**94.81** / 77.75	89.64 / 78.58	**87.59** / **78.63**	95.74 / 82.50	**91.09** / 78.80	**91.02** / **80.53**

gain) and Adobe Firefly Image 5 (+5.88%). In contrast, performance degrades for Reve Image 1.0 (−8.06%) and Z Image Turbo (−3.23%).

Finally, Task 4 achieves the peak average AUC across all benchmarks, representing an improvement of +7.36% over Task 0. On AIGenImages2026, the proposed framework effectively adapts the RINE model, reaching a substantial +20.14% accuracy gain. Overall, these results confirm that the framework enables continual adaptation to evolving data distributions with only minor fluctuations, while mitigating catastrophic forgetting across both generators and in-the-wild data.

5.5 Ablation Studies

Component contribution analysis. We conduct an ablation study on WildFC, AIGenImages2026, and the replay mechanism (see Table 3), using a RINE model pretrained on LDM and Twigma as the baseline. Training on WildFC without replay improves overall performance by +5.03% AUC and +3.22% ACC, but reduces Chameleon ACC by −5.2%. When trained with replay, WildFC consistently improves in-the-wild performance, including +1.44% on Chameleon, +2.03% on Deepfake-Eval, +3.65% on ITW-SM, and +4.48% on MediaEval-ITW, while also generalizing to recent generators, with gains of +6.88% on AIGenImages2026, +1.09% on MNW, +11.85% on Echo-4o, and +8.84% on Nano-Consistent in ACC, despite no access to generator-specific training data. In contrast, training on AIGenImages2026 with replay substantially improves generator performance, yielding +19.49% on AIGenImages2026 and +7.34% on MNW, but degrades in-the-wild generalization, with drops of −3.70% on Chameleon, −2.51% on ITW-SM, and −0.86% on MediaEval-ITW, indicating a specialization bias. Combining both datasets without replay further strengthens generator performance, including +15.56% on Echo-4o and +13.95% on Synthbuster, but significantly harms in-the-wild robustness, with decreases of −13.36% on Chameleon and −4.07% on ITW-SM, highlighting distribution interference between real and synthetic data. Importantly, the replay buffer is crucial for mitigating catastrophic forgetting. A 3% buffer improves performance across all in-the-wild datasets while also increasing performance on AIGenImages2026 by +4.29% and MNW by +4.68%. The 5% buffer yields the best trade-off, recovering in-the-wild performance by +12.34% on Chameleon and +2.18% on Deepfake-Eval, and achieving improvements of +7.36% AUC and +8.01% ACC over the baseline. Increasing the buffer to 10% provides no meaningful gain, with changes of −1.42% ACC and +0.12% AUC, so we adopt 5% for efficiency. Overall, in-the-wild data improves robustness, synthetic data enhances specialization, and replay enables their effective integration.

Effect of training data size. We analyze the effect of training data size by progressively increasing the proportion of AIGenImages2026 and WildFC used to train the RINE model, as shown in Table 4. Even 10% of the data yields substantial improvements, with gains of +11.55% on Echo-4o, +3.68% on Nano-Consistent, and +3.35% on average AUC. At 30%, the model achieves strong performance on generator datasets, with +6.71% on MNW and +24.2% on Echo-4o, while showing small improvements on in-the-wild data with +2.92% on Chameleon and +2.68% on Deepfake-Eval in AUC. However, this comes with instability in in-the-wild ACC, indicating suboptimal calibration rather than reduced discriminative power. Increasing the data to 50% improves stability and raises average accuracy by +3.05% over the 30% setting. Using 100% of the data yields the best performance. Overall, these results suggest that the model adapts to new distributions even with limited data, while larger training sets consistently improve performance.

6 Limitations and Future Work

While our approach achieves strong performance several limitations remain. The fact-check retrieval pipeline relies on weak supervision and web-sourced data, which inevitably introduces noise. Incorrect labeling may affect dataset quality and the generalization ability of the model. Future work should focus on refined curation and scaling data collection efficiency. Furthermore, while we treat generator-driven and in-the-wild data as separate sources, developing principled strategies to integrate them could better approximate real-world distributions. Extensions could explore automated image discovery for emerging models, more advanced continual learning strategies, and multimodal content.

7 Conclusion

In this paper, we introduce a continual adaptive framework for AID, addressing the challenge of distribution shift in a rapidly evolving generative content. Unlike prior approaches that rely on static datasets or generator-specific adaptation, our method models AID as a dynamic problem and enables continuous adaptation. To support this, we propose a fact-check retrieval pipeline that enables automated, weakly supervised dataset construction from real-world sources. We further introduce an evolving dataset designed to reflect both emerging generative models and real-world data distributions. Extensive experiments across multiple benchmarks and two different detection architectures demonstrate the general applicability of our framework, consistently achieving state-of-the-art performance. Our results show that combining in-the-wild and generator-driven data, together with a continual data collection and learning strategy, significantly improves generalization to both recent generative models and real-world scenarios.

Acknowledgments

This work received funding by the Horizon Europe projects AI-CODE (GA no. 101135437) and ELIAS (GA no. 101120237).

References

[1] Aref Azizpour, Tai D Nguyen, Manil Shrestha, Kaidi Xu, Edward Kim, and Matthew C Stamm. 2024. E3: Ensemble of expert embedders for adapting synthetic image detectors to new generators using limited data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 4334–4344.

[2] Quentin Bammey. 2023. Synthbuster: Towards detection of diffusion model generated images. *IEEE Open Journal of Signal Processing* 5 (2023), 1–9.

[3] Bar Cavia, Eliahu Horwitz, Tal Reiss, and Yedid Hoshen. 2024. Real-time deepfake detection in the real-world. *arXiv preprint arXiv:2406.09398* (2024).

[4] Nuria Alina Chandra, Ryan Murtfeldt, Lin Qiu, Arnab Karmakar, Hannah Lee, Emmanuel Tanumihardja, Kevin Farhat, Ben Caffee, Sejin Paik, Changyeon Lee, et al. 2025. Deepfake-eval-2024: A multi-modal in-the-wild benchmark of deepfakes circulated in 2024. *arXiv preprint arXiv:2503.02857* (2025).

[5] Yiqun Chen and James Y Zou. 2023. Twigma: A dataset of ai-generated images with metadata from twitter. *Advances in Neural Information Processing Systems* 36 (2023), 37748–37760.

[6] Zhihong Chen, Xuehai Bai, Yang Shi, Chaoyou Fu, Huanyu Zhang, Haotian Wang, Xiaoyan Sun, Zhang Zhang, Liang Wang, Yuanxing Zhang, et al. 2025. Opengpt-4o-image: A comprehensive dataset for advanced image generation and editing. *arXiv preprint arXiv:2509.24900* (2025).

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

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

[9] William Doherty, Anton Lee, and Heitor Murilo Gomes. 2024. CLOFAI: A Dataset of Real And Fake Image Classification Tasks for Continual Learning. In *International Conference on Neural Information Processing*. Springer, 348–362.

[10] David C Epstein, Ishan Jain, Oliver Wang, and Richard Zhang. 2023. Online detection of ai-generated images. In *Proceedings of the IEEE/CVF international conference on computer vision*. 382–392.

[11] Faehrmann, Mike. 2026. gallery-dl. https://github.com/mikf/gallery-dl Accessed: 5 Apr. 2026.

[12] FAL.ai. 2026. FAL.ai API for Generative Image Models. https://fal.ai Accessed: 4 Apr. 2026.

[13] Google. 2026. Google Fact Check Tools API. https://developers.google.com/fact-check/tools/api Accessed: 4 Apr. 2026.

[14] Fabrizio Guillaro, Giada Zingarini, Ben Usman, Avneesh Sud, Davide Cozzolino, and Luisa Verdoliva. 2025. A bias-free training paradigm for more general ai-generated image detection. In *Proceedings of the Computer Vision and Pattern Recognition Conference*. 18685–18694.

[15] Dimitrios Karageogiou, Quentin Bammey, Valentin Porcellini, Bertrand Goupil, Denis Teyssou, and Symeon Papadopoulos. 2024. Evolution of detection performance throughout the online lifespan of synthetic images. In *European Conference on Computer Vision*. Springer, 400–417.

[16] Dimitrios Karageorgiou, Symeon Papadopoulos, Ioannis Kompatsiaris, and Efstratios Gavves. 2025. Any-resolution ai-generated image detection by spectral learning. In *Proceedings of the Computer Vision and Pattern Recognition Conference*. 18706–18717.

[17] Despina Konstantinidou, Dimitrios Karageorgiou, Christos Koutlis, Olga Papadopoulou, Emmanouil Schinas, and Symeon Papadopoulos. 2025. Navigating the Challenges of AI-Generated Image Detection in the Wild: What Truly Matters? *arXiv preprint arXiv:2507.10236* (2025).

[18] Christos Koutlis and Symeon Papadopoulos. 2024. Leveraging representations from intermediate encoder-blocks for synthetic image detection. In *European Conference on computer vision*. Springer, 394–411.

[19] Chunxiao Li, Xiaoxiao Wang, Meiling Li, Boming Miao, Peng Sun, Yunjian Zhang, Xiangyang Ji, and Yao Zhu. 2025. Bridging the Gap Between Ideal and Real-world Evaluation: Benchmarking AI-Generated Image Detection in Challenging Scenarios. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 20379–20389.

[20] Ouxiang Li, Jiayin Cai, Yanbin Hao, Xiaolong Jiang, Yao Hu, and Fuli Feng. 2025. Improving synthetic image detection towards generalization: An image transformation perspective. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 1*. 2405–2414.

[21] Ziqiang Li, Jiazhen Yan, Ziwen He, Kai Zeng, Weiwei Jiang, Lizhi Xiong, and Zhangjie Fu. 2025. Is artificial intelligence generated image detection a solved problem? *arXiv preprint arXiv:2505.12335* (2025).

[22] Fuxiao Liu, Yinghan Wang, Tianlu Wang, and Vicente Ordonez. 2021. Visual news: Benchmark and challenges in news image captioning. In *Proceedings of the 2021 conference on empirical methods in natural language processing*. 6761–6771.

[23] Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Qing Jiang, Chunyuan Li, Jianwei Yang, Hang Su, et al. 2024. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. In *European conference on computer vision*. Springer, 38–55.

[24] Jiajie Lu, Zhenkan Fu, Na Zhao, Long Xing, Kejiang Chen, Weiming Zhang, and Nenghai Yu. 2025. LiteUpdate: A Lightweight Framework for Updating AI-Generated Image Detectors. *arXiv preprint arXiv:2511.07192* (2025).

[25] Jose G Moreno-Torres, Troy Raeder, Rocío Alaiz-Rodríguez, Nitesh V Chawla, and Francisco Herrera. 2012. A unifying view on dataset shift in classification. *Pattern recognition* 45, 1 (2012), 521–530.

[26] NewsAPI. 2026. NewsAPI. https://newsapi.org/ Accessed: 5 Apr. 2026.

[27] Ontotext. 2026. The Database of Known Fakes (DBKF). https://dbkf.ontotext.com Accessed: 4 Apr. 2026.

[28] OpenAI. 2026. Introducing GPT-5.2. https://openai.com/index/introducing-gpt-5-2/ Accessed: 4 Apr. 2026.

[29] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. 2023. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193* (2023).

[30] Olga Papadopoulou, Manos Schinas, Riccardo Corvi, Dimitrios Karageorgiou, Christos Koutlis, Fabrizio Guillaro, Efstratios Gavves, Hannes Mareen, Luisa Verdoliva, and Symeon Papadopoulos. 2025. Synthetic images at mediaeval 2025: Advancing detection of generative ai in real-world online images. In *Proceedings of the MediaEval 2025 Workshop, Dublin, Ireland and Online*. 25–26.

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

[32] Lorenzo Pellegrini, Davide Cozzolino, Serafino Pandolfini, Davide Maltoni, Matteo Ferrara, Luisa Verdoliva, Marco Prati, and Marco Ramilli. 2025. AI-GenBench: A New Ongoing Benchmark for AI-Generated Image Detection. In *2025 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 1–9.

[33] Lorenzo Pellegrini, Serafino Pandolfini, Davide Maltoni, Matteo Ferrara, Marco Prati, and Marco Ramilli. 2025. Generalized Design Choices for Deepfake Detectors. *arXiv preprint arXiv:2511.21507* (2025).

[34] Qwen Team. 2025. Qwen2.5-VL. https://qwenlm.github.io/blog/qwen2.5-vl/ Accessed: 4 Apr. 2026.

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

[36] Thomas Roca, Marco Postiglione, Chongyang Gao, Isabel Gortner, Zuzanna Wojciak, Pengce Wang, Mahsa Alimardani, Shirin Anlen, Kevin White, Juan Lavista Ferres, Sarit Kraus, Sam Gregory, and V. S. Subrahmanian. 2025. Introducing the MNW benchmark for AI forensics. arXiv:.....

[37] Manos Schinas and Symeon Papadopoulos. 2024. SIDBench: A Python framework for reliably assessing synthetic image detection methods. In *Proceedings of the 3rd ACM International Workshop on Multimedia AI against Disinformation*. 55–64.

[38] Sandeep Singh Sengar, Affan Bin Hasan, Sanjay Kumar, and Fiona Carroll. 2025. Generative artificial intelligence: a systematic review and applications. *Multimedia Tools and Applications* 84, 21 (2025), 23661–23700.

[39] Chuangchuang Tan, Yao Zhao, Shikui Wei, Guanghua Gu, Ping Liu, and Yunchao Wei. 2024. 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*. 28130–28139.

[40] Qwen Team. 2025. Qwen3 Technical Report. arXiv:2505.09388 [cs.CL] https://arxiv.org/abs/2505.09388

[41] UncleCode. 2024. Crawl4AI: An Open-Source LLM-Friendly Web Crawler and Scraper. https://github.com/unclecode/crawl4ai Accessed: 4 Apr. 2026.

[42] Sheng-Yu Wang, Oliver Wang, Richard Zhang, Andrew Owens, and Alexei A Efros. 2020. CNN-generated images are surprisingly easy to spot... for now. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 8695–8704.

[43] Yabin Wang, Zhiwu Huang, and Xiaopeng Hong. 2022. S-prompts learning with pre-trained transformers: An occam's razor for domain incremental learning. *Advances in Neural Information Processing Systems* 35 (2022), 5682–5695.

[44] Shilin Yan, Ouxiang Li, Jiayin Cai, Yanbin Hao, Xiaolong Jiang, Yao Hu, and Weidi Xie. 2025. A Sanity Check for AI-generated Image Detection. In *Proceedings of the International Conference on Learning Representations (ICLR)*.

[45] Shamim Yazdani, Akansha Singh, Nripsuta Saxena, Zichong Wang, Avash Palikhe, Deng Pan, Umapada Pal, Jie Yang, and Wenbin Zhang. 2025. Generative AI in depth: A survey of recent advances, model variants, and real-world applications. *Journal of Big Data* 12, 1 (2025), 230.

[46] Junyan Ye, Dongzhi Jiang, Zihao Wang, Leqi Zhu, Zhenghao Hu, Zilong Huang, Jun He, Zhiyuan Yan, Jinghua Yu, Hongsheng Li, Conghui He, and Weijia Li. 2025. Echo-4o: Harnessing the power of gpt-4o synthetic images for improved image generation. *arXiv preprint arXiv:2508.09987* (2025).

[47] Junyan Ye, Leiqi Zhu, Yuncheng Guo, Dongzhi Jiang, Zilong Huang, Yifan Zhang, Zhiyuan Yan, Haohuan Fu, Conghui He, and Weijia Li. 2025. Realgen: Photo-realistic text-to-image generation via detector-guided rewards. *arXiv preprint arXiv:2512.00473* (2025).

[48] Hanzhe Yu, Yun Ye, Jintao Rong, Qi Xuan, and Chen Ma. 2025. RealHD: A High-Quality Dataset for Robust Detection of State-of-the-Art AI-Generated Images. In *Proceedings of the 33rd ACM International Conference on Multimedia*. 11394–11403.