FORENSIC PROXY GENERATION: ADAPTING DEEPFAKE DETECTORS TO BLACK-BOX GENERATORS

Anonymous authors
Paper under double-blind review

ABSTRACT

As commercial image generators rapidly evolve, deepfake detectors must continually adapt to the images they produce, typically through retraining or fine-tuning on newly generated images. However, collecting a large-scale dataset from each new generator can be impractical when only black-box access is available and query costs or usage restrictions limit extensive collection or downstream reuse. To this end, we introduce **F**orensic **P**roxy **G**eneration (FPG), a two-stage framework that converts limited black-box access to a target generator into reusable proxy images for detector training. First, we treat the target generator as a teacher and distill teacher-relevant generative characteristics into a student generator. Second, we refine student-generated images through detector-response alignment. Under our problem setting, the resulting proxy images can be used for detector training while reducing the need for large-scale teacher querying and direct reuse of teacher-generated images. Across six target generators, FPG improves detection of teacher-generated images and balanced real–fake data without directly using teacher outputs for detector training. Under limited access, FPG expands a small number of teacher queries into proxy data that can outperform direct training on teacher outputs. These results show that FPG effectively adapts deepfake detectors to new generators under access and reuse constraints. The code is available at https://anonymous.4open.science/r/hci-code-3B8E.

1 INTRODUCTION

Recent advances in generative modeling have enabled increasingly realistic AI-generated images (Ho et al., 2020; Rombach et al., 2022; Esser et al., 2024), while commercial image-generation tools have made advanced image synthesis widely accessible (Raisinghani, 2025). This is particularly concerning for facial imagery, where deepfakes can be used to impersonate individuals, spread misinformation, and undermine trust in visual content (Mustak et al., 2023). These risks have motivated extensive research on detecting AI-generated facial images (Li et al., 2020; Shiohara & Yamasaki, 2022; Ojha et al., 2023), yet detectors trained on existing data may not generalize reliably to images from newly introduced generators.

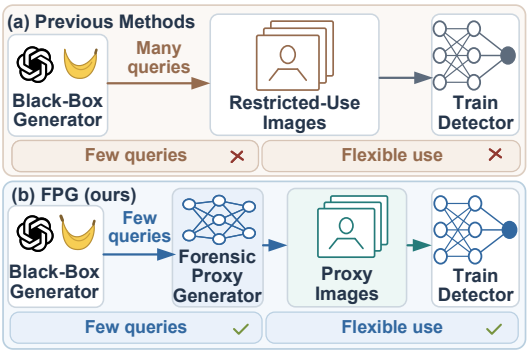

Figure 1: *Comparison of conventional adaptation and FPG. FPG replaces extensive target queries with reusable student-generated proxy images.*

A straightforward way to adapt a detector to such a generator is to collect its outputs and use them for retraining or fine-tuning. However, this approach assumes sufficient access to images from the target generator and the ability to reuse them for downstream training. For leading commercial systems such as Google's Nano Banana Pro (Google DeepMind, 2025), large-scale collection incurs direct per-image query costs (Google, 2026a), while provider-specific terms may restrict model development or downstream reuse (Google, 2026b). Together, these **access** and **usage** constraints make detector adaptation more difficult and introduce uncertainty as permitted downstream uses

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

may change over time. Therefore, as image generators continue to evolve, keeping detector training data up to date under these constraints can become costly, time-consuming, and difficult to sustain.

Existing approaches only partially address this problem setting. Generative distillation methods such as DMD (Yin et al., 2024b) and DMD2 (Yin et al., 2024a) offer one way to transfer information from a target generator into an accessible model, but require white-box access to the target generator and are optimized for image synthesis rather than downstream detection. Dataset-mixing approaches provide another way to approximate the target domain using existing data, yet may fail to capture target-generator-specific artifacts and are not explicitly optimized for detection (Zhao et al., 2020). These limitations reveal a fundamental **mismatch between generation and detection**. We therefore study how to adapt deepfake detectors to new generators with only limited black-box access, without directly using their outputs as detector-training examples.

To address these challenges, we introduce **Forensic Proxy Generation (FPG)**, a two-stage framework that converts limited access to a target generator into reusable proxy images for detector training. As illustrated in Fig. 1, FPG reduces target-generator querying by transferring target-specific information into an accessible student generator. In Stage 1, FPG treats the target generator as a teacher and distills teacher-relevant generative characteristics into the student. In Stage 2, it refines student-generated images by aligning their detector responses with those of teacher-generated images. The resulting proxy images can then be generated without further teacher queries and used to adapt detectors to the target generator. Our contributions can be summarized as follows:

- We formulate a **new problem** of adapting deepfake detectors to black-box generators under **access** and **reuse** constraints: the target generator can be queried only a limited number of times, and its outputs cannot be used directly as detector-training examples.
- We propose FPG, a two-stage framework that transfers teacher-specific information to a student generator through teacher-guided latent and frequency alignment, and then refines its outputs through detector-response alignment to produce reusable proxy images.
- We validate FPG through extensive experiments across six closed- and open-source target generators. FPG improves detection of teacher-generated images and balanced real–fake data without directly using teacher outputs for detector training. Under limited access, it expands a small number of teacher queries into proxy data that can outperform direct training on teacher outputs.

2 Related Work

AI-Generated Image Detection. AI image generation has advanced from generative adversarial networks (Goodfellow et al., 2014) to diffusion models (Ho et al., 2020; Rombach et al., 2022) and rectified-flow models (Esser et al., 2024), enabling increasingly realistic and diverse image synthesis. In parallel, substantial work has focused on improving the generalization of AI-generated image detectors across architectures, datasets, and generative models. Common backbones include Xception (Chollet, 2017), EfficientNet-B4 (Tan & Le, 2019), SwinV2-Tiny (Liu et al., 2022), and CLIP ViT-B/16 (Radford et al., 2021). More recent approaches improve generalization by learning transferable representations or exploiting forensic artifacts. Effort (Yan et al., 2025), IAPL (Li et al., 2026), and LTD (Yang et al., 2026) use orthogonal feature adaptation, image-adaptive prompts, and inter-layer consistency, respectively. FerretNet-S (Liang et al., 2025), LOTA (Wang et al., 2025), and MLEP (Yuan et al., 2025) exploit local pixel dependencies, bit-plane noise, and local entropy, while Stay-Positive (Sundara Rajan & Lee, 2025) reduces reliance on real-image features and PGC (Zhou et al., 2026) calibrates predictions using local peak evidence. Our work instead focuses on adapting detectors to a new target generator under limited access.

Model Distillation and Dataset Mixing. Model distillation transfers knowledge from a teacher model to a student model (Hinton et al., 2015). Diffusion distillation methods such as DMD (Yin et al., 2024b) and DMD2 (Yin et al., 2024a) distill diffusion teachers into efficient student generators using access to the teacher's internal denoising or score predictions. Reference-based adaptation provides another way to transfer target-specific information into a generative model. DreamBooth (Ruiz et al., 2023) and DomainGallery (Duan et al., 2024) adapt pretrained generators to a specific subject or domain using a small set of reference images. Dataset mixing offers a complementary data-level strategy by selecting or combining existing source data to better match a target domain. For example, MDDA (Zhao et al., 2020) selects source samples relevant to the target and aggregates predictions from source classifiers. FPG instead uses limited black-box supervision from the target generator to construct reusable proxy images that are explicitly optimized for downstream detector adaptation.

3 Problem Setup and Motivation

Problem Setup. Let $\mathcal{A}$ denote a permissible dataset for deepfake detection, where each $(x, y) \in \mathcal{A}$ contains an image x and label $y \in \{0, 1\}$ indicating whether it is real or generated. We also assume access to a permissible set $\mathcal{P}$ of image–prompt pairs (x, c). We consider a target generative model T, referred to as the teacher generator, accessible only through black-box queries. Given a prompt c, the teacher produces an image $x^T \sim T(\cdot \mid c)$. We assume a query budget of N, with no access to the teacher's model weights or gradients. Under this budget, we select N image–prompt pairs (x_i, c_i) from $\mathcal{P}$ and query T with each prompt c_i, producing $x_i^T \sim T(\cdot \mid c_i)$. These queries form the teacher set $\mathcal{I}_T = \{(x_i, c_i, x_i^T)\}_{i=1}^N$. Let D denote a deepfake detector trained on $\mathcal{A}$, with $p_D(x) \in [0, 1]$ being the predicted probability that image x is generated. Because $\mathcal{A}$ may not reflect the distribution or generator-specific artifacts of T, D may not generalize reliably to teacher-generated images. We assume that images in $\mathcal{I}_T$ may be used as limited supervision but not as downstream detector-training examples. This is a problem-setting constraint rather than a claim about any specific provider policy. Our goal is to improve detection of images generated by T under these access and reuse assumptions.

Motivation for Proxy Generation. The limited size of $\mathcal{I}_T$ and the restriction against using teacher-generated images for downstream detector training make it difficult to obtain sufficient target-specific training data from the teacher. A natural solution is therefore to use the limited teacher supervision to transfer target-specific information into an accessible student generator S. Under our problem setting, images generated by S may be used for detector training, as they are not teacher outputs. Once trained, S can generate additional proxy images without further queries to the teacher.

Limitations of Generative Distillation. Generative distillation provides a natural mechanism for learning a student proxy from teacher supervision, but may not preserve detection-relevant forensic characteristics. We therefore examine whether a student S learned through conventional generative distillation preserves teacher-specific spectral characteristics. We instantiate T with Nano Banana Pro (Google DeepMind, 2025) and construct $\mathcal{I}_T$ from $N = 1{,}000$ unique prompts. Using $\mathcal{I}_T$ as black-box supervision, we train a Stable Diffusion 3 Medium (Esser et al., 2024) student S with a DMD2-inspired (Yin et al., 2024a) baseline (Appendix A.1.1) and compare its outputs with those of T. Using the same prompts as $\mathcal{I}_T$, we generate a student set $\mathcal{I}_S = \{x_i^S\}_{i=1}^N$, where $x_i^S \sim S(\cdot \mid c_i)$.

Frequency-domain artifacts provide useful forensic cues for detecting AI-generated images (Frank et al., 2020; Karageorgiou et al., 2025). We therefore compare teacher and student outputs by computing log-amplitude Fourier spectra for each image and averaging across samples. As shown in Fig. 2, the distilled student differs from the teacher in the frequency domain, with a cosine similarity of 0.917, indicating that generative distillation alone does not fully preserve teacher-specific spectral characteristics. This motivates explicit frequency alignment during student training. Applying this alignment yields student outputs $\mathcal{I}_S'$ whose cosine similarity to the teacher outputs increases to 0.948; we later formalize this frequency alignment in Sec. 4.2. Although frequency alignment reduces the spectral gap between teacher and student outputs, it does not ensure that they produce similar responses under a downstream detector. This analysis motivates a two-level strategy that aligns the student with the teacher at both the *generative* and *detector* levels.

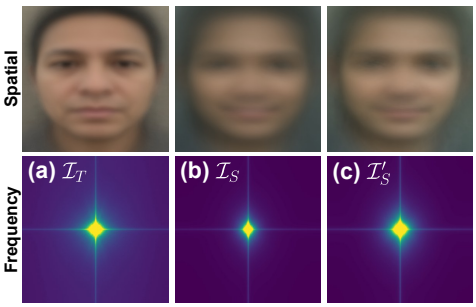

Figure 2: *Averaged spatial-domain images (top) and frequency spectra (bottom) for teacher and student outputs. Cosine similarity to the teacher outputs in (a) increases from 0.917 for (b) to 0.948 for (c).*

4 Method

4.1 Framework Overview

Following this motivation, FPG is a two-stage framework illustrated in Fig. 3. Given $\mathcal{P}$ and teacher supervision $\mathcal{I}_T$, Stage 1 trains the student generator S to capture teacher-relevant generative characteristics through latent and frequency alignment. Stage 2 refines student outputs by aligning their detector responses with those of the corresponding teacher outputs under the frozen detector D.

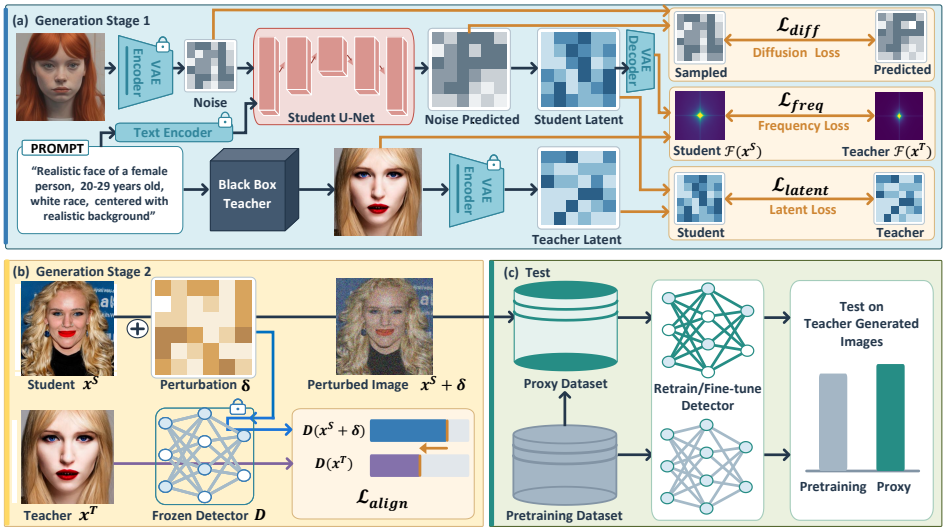

Figure 3: *Overview of the proposed FPG framework.*

Under our problem setting, the resulting student-generated images can be used for downstream detector training and expanded without additional teacher queries. We use them to construct a proxy dataset $\mathcal{B}_S$ of labeled fake images that can be combined with permissible detector data $\mathcal{A}$. Detectors trained with these proxy images are thus exposed to teacher-relevant information without directly training on images generated by T. We next formalize each stage and its corresponding objective.

4.2 Stage 1: Black-Box Generator Distillation

In Stage 1, we train the student S using permissible image–prompt pairs together with the teacher supervision in $\mathcal{I}_T$. We present the formulation for a latent diffusion student, as illustrated in Fig. 3, whose native generative objective is augmented with teacher-guided latent and frequency alignment.

Diffusion Reconstruction. For $(x, c) \in \mathcal{P}$, we map x to a latent $z_0 = E(x)$ using a frozen variational autoencoder (VAE) E. We then sample a timestep t and Gaussian noise $\epsilon \sim \mathcal{N}(0, I)$ and construct the noisy latent $z_t = \alpha_t z_0 + \sigma_t \epsilon$, where α_t and σ_t are set by the diffusion noise schedule. Given z_t, the student predicts the added noise as $\epsilon_S(z_t, t, c)$. The diffusion reconstruction loss is

$$\mathcal{L}_{\text{diff}} = \mathbb{E}_{(x,c)\sim\mathcal{P},\, t,\, \epsilon} \left\| \epsilon - \epsilon_S(z_t, t, c) \right\|_2^2 . \tag{1}$$

This objective preserves the student's native diffusion training on permissible data.

Teacher-Guided Latent Alignment. The diffusion reconstruction loss alone does not incorporate teacher-specific information. We therefore introduce a teacher-guided latent alignment loss using the black-box supervision in $\mathcal{I}_T$. For $(x_i, c_i, x_i^T) \in \mathcal{I}_T$, we encode the teacher output as $z_{0,i}^T = E(x_i^T)$. Following the same noising process, we construct $z_{t,i}$ from x_i and recover the student's estimate of the clean latent, $\hat{z}_{0,i}^S = \frac{z_{t,i} - \sigma_t \epsilon_S(z_{t,i}, t, c_i)}{\alpha_t}$. We then align this estimate with the teacher latent using

$$\mathcal{L}_{\text{latent}} = \mathbb{E}_{(x_i,c_i,x_i^T)\sim\mathcal{I}_T,\, t,\, \epsilon} \left\| \hat{z}_{0,i}^S - z_{0,i}^T \right\|_2^2 . \tag{2}$$

Teacher-Guided Frequency Alignment. Motivated by the spectral discrepancy identified in Sec. 3, we introduce a teacher-guided frequency alignment loss that aligns the frequency-domain characteristics of student and teacher outputs. For $(x_i, c_i, x_i^T) \in \mathcal{I}_T$, the student generates $x_i^S \sim S(\cdot \mid c_i)$. Let $\mathcal{F}(x)$ denote the log-amplitude Fourier spectrum of the luminance channel of x. We define

$$\mathcal{L}_{\text{freq}} = \mathbb{E}_{(x_i,c_i,x_i^T)\sim\mathcal{I}_T} \left\| \mathcal{F}(x_i^S) - \mathcal{F}(x_i^T) \right\|_2^2 . \tag{3}$$

Further details, including DC exclusion and frequency averaging, are provided in Appendix A.1.2.

The Stage 1 learning objective is $\mathcal{L}_{\text{stage1}} = \mathcal{L}_{\text{diff}} + \lambda_{\text{latent}} \mathcal{L}_{\text{latent}} + \lambda_{\text{freq}} \mathcal{L}_{\text{freq}}$, where λ_{latent} and λ_{freq} are trade-off hyperparameters. Although Stage 1 is formulated for a latent diffusion student, it supports alternative architectures and objectives. A flow-matching student can replace $\mathcal{L}_{\text{diff}}$ with its flow-matching objective while retaining the teacher-guided latent and frequency alignment losses; details are provided in Appendix A.1.3. Our experiments in Sec. 5 demonstrate this flexibility.

4.3 Stage 2: Detector-Response Alignment

As discussed in Sec. 3, generative alignment alone does not ensure similar detector responses for student and teacher outputs. Stage 2 therefore directly aligns student outputs by matching their responses more closely to those of the corresponding teacher outputs under the frozen detector D.

For $(x_i, c_i, x_i^T) \in \mathcal{I}_T$, the student generates $x_i^S \sim S(\cdot \mid c_i)$ using the same prompt, requiring no additional queries to T. We seek a small perturbation δ that brings the detector response of $x_i^S + \delta$ closer to that of x_i^T while preserving the original student output as much as possible. We optimize

$$\mathcal{L}_{\text{align}}(\delta) = \frac{1}{2}\|\delta\|_2^2 + \lambda \left| p_D(x_i^S + \delta) - p_D(x_i^T) \right|, \tag{4}$$

where λ is a trade-off hyperparameter for detector-response alignment. To enforce a hard bound on the perturbation size, we restrict δ to an ℓ_2 ball of radius ϵ and optimize it with projected gradient descent (PGD), initialized at $\delta^{(0)} = 0$. Each iteration takes a gradient descent step on $\mathcal{L}_{\text{align}}$, followed by projection back onto the constraint set: $\delta^{(k+1)} = \Pi_{\|\delta\|_2 \le \epsilon}\left(\delta^{(k)} - \eta \nabla_\delta \mathcal{L}_{\text{align}}(\delta^{(k)})\right)$, where η is the step size and $\Pi_{\|\delta\|_2 \le \epsilon}$ denotes projection onto the ℓ_2 ball.

After optimization, the aligned image $\tilde{x}_i^S = x_i^S + \delta^*$ yields a detector response closer to that of x_i^T. These aligned images form the proxy dataset $\mathcal{B}_S = \{(\tilde{x}_j^S, 1)\}_{j=1}^M$ for downstream detector training. For each $(x_i, c_i, x_i^T) \in \mathcal{I}_T$, multiple student outputs can be sampled from $S(\cdot \mid c_i)$ and individually aligned to x_i^T, allowing M to exceed the query budget N without further queries to T.

Together, the two stages of FPG transfer teacher information at complementary levels: Stage 1 captures generative characteristics, while Stage 2 aligns detector responses. The resulting proxy dataset converts limited black-box access to T into reusable supervision for downstream detector training, enabling generalization to teacher outputs without directly including them in the final training set.

5 Experiments

5.1 Settings

Datasets. We evaluate FPG across six target-generator settings spanning closed- and open-source generators. We use LAION-Face (Zheng et al., 2022) as the permissible image–prompt set $\mathcal{P}$ for student training. For closed-source generators, ***NanoSet*** (Nano Banana Pro + AI-Face) uses 10,300 real and 10,300 fake images from AI-Face (Lin et al., 2025) as the permissible data $\mathcal{A}$, with 1,000 faces generated by Nano Banana Pro (Google DeepMind, 2025) as teacher supervision and 430 faces reserved for evaluation. ***HiggsfieldSet*** (Higgsfield Soul v2 Standard + AI-Face), ***GrokSet*** (Grok Imagine + AI-Face), and ***GPTSet*** (GPT Image 2 + AI-Face) use the same AI-Face data and contain 600 teacher-generated images for supervision and 400 for evaluation from Higgsfield Soul v2 Standard (Higgsfield, 2026), Grok Imagine (xAI, 2026), and GPT Image 2 (OpenAI, 2026), respectively. Teacher-generated images for these four datasets were collected from February 10 to June 11, 2026; see Appendix A.4 for details. For open-source generators, ***CollabSet*** (Collaborative Diffusion + FF++) uses 12,926 real and 63,213 fake images from FaceForensics++ (FF++) (Rossler et al., 2019) as the permissible data $\mathcal{A}$, with 18,000 faces generated by Collaborative Diffusion (Huang et al., 2023) as teacher supervision and 954 faces reserved for evaluation. ***IDiffSet*** (IDiff-Face + FF++) uses the same FF++ data and an 18,000/946 split, with fakes from IDiff-Face (Boutros et al., 2023). For consistency with our problem setting, we treat these open-source generators as black boxes.

Evaluation Metrics. We consider two settings: (i) ***teacher-only***, containing only teacher-generated images, and (ii) ***mixed***, containing equal numbers of teacher-generated images and real images from AI-Face or FF++. All evaluation images are disjoint from the permissible data $\mathcal{A}$ and teacher supervision $\mathcal{I}_T$. We report T-ACC for teacher-only and ACC, AUC, EER, and FAR for mixed evaluation. T-ACC, ACC, and FAR use a threshold of 0.5, while AUC and EER use prediction scores. Higher T-ACC, ACC, and AUC are better; lower EER and FAR are better. All metrics are reported in %.

Baselines. Because no existing method operates under the same constraints, we consider two eligible alternatives. (i) **Base** trains the detector using only permissible real and fake images, without information from the target generator. (ii) **MDDA-inspired** (Zhao et al., 2020) is a dataset-mixing baseline that adapts MDDA's target-relevant sample-selection component to our setting. Teacher-generated images guide selection of the most target-relevant samples from multiple gen-

Table 1: *Performance across six target-generator settings. Superscript † denotes a closed-source target generator, while superscript ⋆ denotes an open-source target generator.*

Method	NanoSet†					HiggsfieldSet†					GrokSet†				
	T-ACC↑	AUC↑	ACC↑	EER↓	FAR↓	T-ACC↑	AUC↑	ACC↑	EER↓	FAR↓	T-ACC↑	AUC↑	ACC↑	EER↓	FAR↓
Base	73.95	84.32	76.63	22.56	26.05	88.50	87.43	81.63	18.75	11.50	90.50	95.79	89.38	11.25	9.50
MDDA-Insp.	80.47	83.44	77.91	23.72	19.53	63.00	87.86	75.63	21.75	37.00	89.25	95.80	89.00	11.25	10.75
FPG (Ours)	**93.49**	**89.61**	**84.07**	**17.91**	**6.51**	**91.50**	**89.47**	**83.25**	**16.50**	**8.50**	**94.75**	**97.75**	**92.00**	**7.50**	**5.25**

Method	GPTSet†					CollabSet⋆					IDiffSet⋆				
	T-ACC↑	AUC↑	ACC↑	EER↓	FAR↓	T-ACC↑	AUC↑	ACC↑	EER↓	FAR↓	T-ACC↑	AUC↑	ACC↑	EER↓	FAR↓
Base	78.25	91.00	81.50	17.50	21.75	67.30	80.27	72.80	25.47	32.70	95.03	87.33	84.20	18.18	4.97
MDDA-Insp.	93.50	91.39	85.50	15.00	6.50	91.72	89.29	83.28	17.82	8.28	99.89	94.94	85.04	11.21	0.11
FPG (Ours)	**93.75**	**92.51**	**85.88**	**14.50**	**6.25**	**98.22**	**94.96**	**87.68**	**12.79**	**1.78**	**100.00**	**96.69**	**85.68**	**8.56**	**0.00**

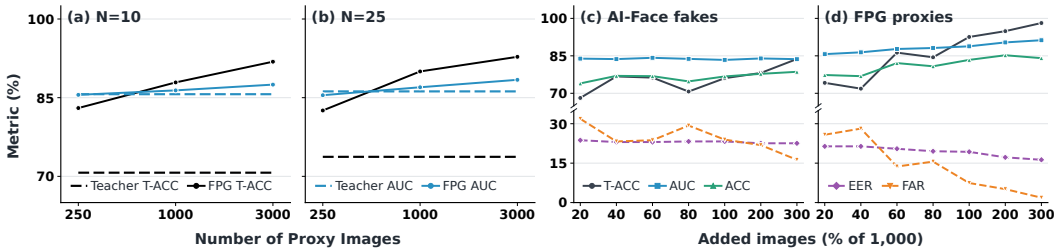

Figure 4: *Effect of limited teacher access and proxy scaling on NanoSet. (a–b) FPG expands $N = 10$ or $N = 25$ teacher queries into proxy training sets without additional teacher access; (c–d) proxy scaling is compared with adding AI-Face fakes under the default $N = 1{,}000$ setting.*

eration sources in AI-Face, but are not used directly for detector training, making this baseline compatible with our constraints; the selected samples are combined with the permissible data.

Detectors. We evaluate 11 AI-generated image detectors from two broad categories: (i) **Backbone Architectures**: Xception (Chollet, 2017), EfficientNet-B4 (Tan & Le, 2019), SwinV2-Tiny (Liu et al., 2022), and CLIP ViT-B/16 (Radford et al., 2021); and (ii) **Specialized Detectors**: Effort (Yan et al., 2025), LTD (Yang et al., 2026), FerretNet-S (Liang et al., 2025), LOTA (Wang et al., 2025), MLEP (Yuan et al., 2025), Stay-Positive (Sundara Rajan & Lee, 2025), and PGC (Zhou et al., 2026). Unless otherwise specified, Xception is used as the default detector.

Implementation Details. Each experiment is run on a single NVIDIA RTX A6000 GPU (48 GB). For closed-source target generators, we use Stable Diffusion 3 Medium (SD3) (Esser et al., 2024) as the student generator, producing 512×512 images with 50 sampling steps. For open-source target generators, we use Stable Diffusion 1.5 (SD1.5) (Rombach et al., 2022). Unless otherwise specified, we generate one proxy image per teacher prompt, yielding $M = N$ proxy images. In Stage 2, we perform 40 PGD steps with step size $\eta = 0.05$ using a frozen detector trained on the permissible data. Unless otherwise specified, we set $\lambda_{\text{latent}} = 0.1$, $\lambda = 1$, and $\epsilon = 1$. We calibrate λ_{freq} at the start of training using up to 32 samples from $\mathcal{I}_T$, setting it to the median gradient-magnitude ratio between $\lambda_{\text{latent}}\mathcal{L}_{\text{latent}}$ and $\mathcal{L}_{\text{freq}}$. The resulting λ_{freq} is fixed throughout training and not tuned on downstream performance. Other optimization hyperparameters, including learning rates and training schedules, are set according to the student architecture, detector architecture, and training objective.

5.2 Main Results

We first answer two questions: (i) ***Reuse Constraint****: can proxy images improve detector generalization* ***without*** *using teacher-generated images for detector training?* (ii) ***Access Constraint****: can proxy images match or surpass limited teacher-generated images when used for detector training?*

Reuse Constraint. Table 1 evaluates whether FPG can improve detector generalization without using teacher-generated images for detector training. Compared with Base, FPG improves AUC by 5.29, 2.04, 1.96, and 1.51 percentage points on the closed-source NanoSet, HiggsfieldSet, GrokSet, and GPTSet, respectively. On NanoSet, T-ACC increases substantially from 73.95% to 93.49%, while AUC rises from 84.32% to 89.61%. We further evaluate FPG on the open-source settings,

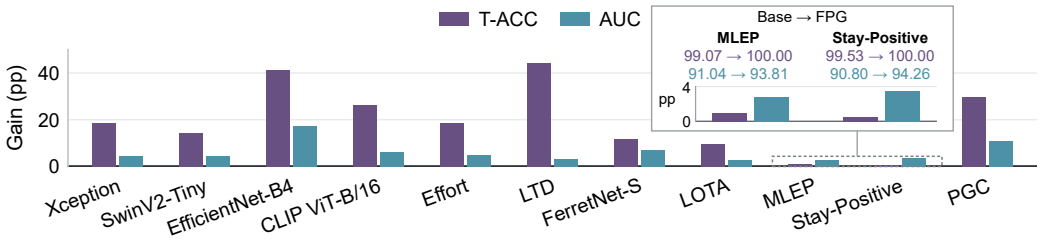

Figure 5: *FPG gains over Base across 11 detectors on NanoSet. pp denotes percentage points.*

Table 2: *Cross-dataset generalization from GPTSet to three unseen target-generator settings.*

Method	GPTSet → CollabSet					GPTSet → IDiffSet					GPTSet → GrokSet				
	T-ACC↑	AUC↑	ACC↑	EER↓	FAR↓	T-ACC↑	AUC↑	ACC↑	EER↓	FAR↓	T-ACC↑	AUC↑	ACC↑	EER↓	FAR↓
Base	92.66	86.59	68.50	20.44	7.34	99.05	94.56	71.67	12.26	0.95	84.25	92.52	84.50	15.25	15.75
MDDA-Insp.	95.28	82.68	58.91	24.11	4.72	97.78	84.30	60.15	22.73	2.22	91.75	90.20	84.63	16.75	8.25
FPG (Ours)	**95.81**	**93.79**	**74.90**	**13.10**	**4.19**	**100.00**	**99.32**	**77.01**	**3.38**	**0.00**	**96.50**	**97.43**	**90.75**	**7.75**	**3.50**

where it improves all reported metrics over Base, indicating that these gains extend across distinct target generators. Across all six settings, FPG also outperforms the MDDA-inspired baseline on every reported metric. Together, these results show that FPG proxies improve detector performance across both closed- and open-source target generators while operating under the reuse constraint.

Access Constraint. We evaluate on NanoSet whether FPG can expand a small teacher-query budget into a larger proxy dataset. Once trained, the student can generate additional proxies without further teacher queries. In Fig. 4(a–b), we set $N = 10$ or $N = 25$ and compare direct training on teacher outputs with increasing numbers of proxies generated from the same supervision. With only 250 proxies, FPG already achieves higher T-ACC in both settings; as the proxy set grows, T-ACC and AUC improve further, with AUC eventually surpassing the corresponding teacher-trained baseline. We then return to the default $N = 1{,}000$ setting to test whether these gains simply reflect adding more fake training data. Fig. 4(c–d) compares randomly sampled AI-Face fakes with FPG proxies at increasing scales. With additional AI-Face fakes, AUC, ACC, and EER remain relatively flat, whereas scaling FPG proxies improves all five metrics. Thus, FPG turns limited teacher access into scalable proxy data that improves downstream detection, thereby addressing the access constraint.

5.3 Extended Study

Applicability Across Detectors. To demonstrate the broad applicability of FPG, we test it on NanoSet across 11 detectors. Each detector is trained on the AI-Face permissible data augmented with FPG proxies and compared with the Base setting. As shown in Fig. 5, FPG improves both T-ACC on teacher-generated images and AUC across all detectors. Notably, MLEP (Yuan et al., 2025) and Stay-Positive (Sundara Rajan & Lee, 2025) already achieve Base T-ACC scores of 99.07% and 99.53%, respectively, leaving little room for improvement. FPG nevertheless increases both to 100% while improving AUC by 2.77 and 3.47 percentage points, respectively. These results show that FPG can consistently enhance adaptation across diverse detector architectures.

Cross-Dataset Performance. Beyond adaptation to the target generator, generalization to previously unseen generators is important for deepfake detection. We train a detector under the GPTSet setting and evaluate it on CollabSet, IDiffSet, and GrokSet. As shown in Table 2, FPG outperforms Base and the MDDA-inspired baseline across all reported metrics in all three settings. For example, on CollabSet, FPG improves AUC from 86.59% to 93.79% and ACC from 68.50% to 74.90% over Base. Notably, the ACC gains on CollabSet and IDiffSet exceed the corresponding T-ACC gains. Because the mixed test set contains equal numbers of real and teacher-generated images, this indicates that FPG also improves real-image classification rather than only unseen-fake detection. Overall, FPG improves cross-generator generalization without sacrificing real-image recognition.

Applicability to Detector Fine-Tuning. Beyond full detector retraining, we evaluate whether FPG proxies can support parameter-efficient fine-tuning on NanoSet. Specifically, we consider two strategies: LoRA (Hu et al., 2022) and Partial Fine-Tuning (FT), where the earlier layers are frozen and only the final two blocks and classification head are updated. We evaluate two training settings: 1,000 proxy fake images alone (*Fake Only*) and the same proxies combined with 1,000 randomly sampled real images from AI-Face to form a balanced set (*Fake + Real*). As shown in Table 3,

Table 3: *Effect of different fine-tuning settings across three detectors on NanoSet.*

Training data	Adaptation method	Xception					EfficientNet-B4					CLIP ViT-B/16				
		T-ACC↑	AUC↑	ACC↑	EER↓	FAR↓	T-ACC↑	AUC↑	ACC↑	EER↓	FAR↓	T-ACC↑	AUC↑	ACC↑	EER↓	FAR↓
	Base	73.95	84.32	76.63	22.56	26.05	3.49	69.47	45.81	34.19	96.51	43.72	84.24	66.51	23.72	56.28
Fake Only	LoRA	**96.51**	84.56	**81.05**	21.86	**3.49**	**7.67**	70.23	46.63	33.95	**92.33**	68.37	86.29	76.40	22.09	31.63
	Partial FT	95.12	84.72	80.70	21.86	4.88	7.21	69.94	46.51	33.95	92.79	60.00	84.77	72.91	23.26	40.00
Fake + Real	LoRA	87.21	**85.33**	80.47	**21.40**	12.79	6.98	**70.30**	**46.86**	**33.49**	93.02	58.14	85.73	72.56	22.33	41.86
	Partial FT	77.44	84.73	77.79	21.86	22.56	7.21	70.17	46.51	33.95	92.79	**78.14**	**86.71**	**78.49**	**21.40**	**21.86**

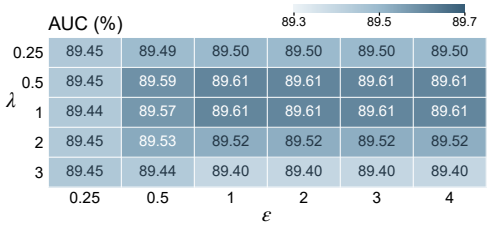

Figure 6: *Sensitivity to hyperparameters.*

Table 4: *Effect of Stage 1 student architecture on downstream detection performance on NanoSet.*

Variant	T-ACC ↑	AUC ↑	ACC ↑	EER ↓	FAR ↓
FPG (VAE)	73.26	84.49	75.93	22.33	26.74
FPG (GAN)	66.05	83.87	72.79	24.19	33.95
FPG (SD1.5)	63.72	83.31	71.74	23.72	36.28
FPG (SD3)	**93.49**	**89.61**	**84.07**	**17.91**	**6.51**

both fine-tuning strategies improve generalization to the target generator across all three detectors and both training settings. Notably, the *Fake Only* setting shows that FPG proxies can adapt detectors without requiring access to the original permissible training data, enabling lightweight updates to pretrained detectors. The *Fake + Real* setting shows that FPG proxies can be incorporated into balanced fine-tuning with real images while still improving generalization to the target generator.

5.4 Sensitivity Analysis

Hyperparameter Sensitivity. To assess the sensitivity of the Stage 2 hyperparameters λ and ϵ, we evaluate $\lambda \in \{0.25, 0.5, 1, 2, 3\}$ and $\epsilon \in \{0.25, 0.5, 1, 2, 3, 4\}$ on NanoSet. As shown in Fig. 6, AUC ranges only from 89.40% to 89.61% across all 30 combinations, a variation of just 0.21 percentage points. These results show that Stage 2 is robust to moderate changes in λ and ϵ.

Student Architecture. We evaluate how the Stage 1 student architecture affects downstream detection on NanoSet. Specifically, we replace SD3 with VAE (Kingma & Welling, 2014), GAN (Goodfellow et al., 2014), and SD1.5. As shown in Table 4, SD3 performs best across all reported metrics. This indicates that student architecture and capability can affect the transfer of teacher-specific generative and forensic characteristics and downstream detector performance. For highly realistic target generators, a more capable student such as SD3 may therefore be preferable.

Detector Dependence Analysis. Stage 2 aligns proxies using a specific detector, which can introduce detector-specific information. To study this effect, we train EfficientNet-B4, CLIP ViT-B/16, and SwinV2-Tiny using proxies guided by either Xception or the detector itself. As shown in Table 5, self-guided proxies outperform Xception-guided proxies across all three detectors. These results show that detector-specific guidance can improve performance.

Table 5: *Detection performance using detector-specific proxies versus Xception-guided proxies.*

Detector	Proxy Source	T-ACC ↑	AUC ↑	ACC ↑	EER ↓	FAR ↓
EfficientNet-B4	Self-guided	**38.37**	**85.66**	**66.98**	**20.00**	**61.63**
	Xception-guided	29.77	81.37	62.67	23.02	70.23
CLIP ViT-B/16	Self-guided	**75.35**	**90.33**	**81.05**	**17.21**	**24.65**
	Xception-guided	70.47	89.42	79.07	18.60	29.53
SwinV2-Tiny	Self-guided	**75.81**	**91.54**	**81.98**	**18.37**	**24.19**
	Xception-guided	72.33	90.17	80.35	19.07	27.67

Effect of Proxy Image Proportion. We compare detector training with only Stage 1 student-generated images (Student), a roughly equal mixture of student-generated and Stage 2 proxy images (Mix), or only proxy images (Proxy), while keeping the training-set size fixed. As shown in Table 6, detection performance generally improves as the proportion of proxy images increases across all three detectors. These results show that Stage 2 proxies are more effective for detector training than Stage 1 student-generated images, with a high proxy proportion generally improving performance.

5.5 Ablation Study

We conduct ablations on NanoSet to assess the contributions of $\mathcal{L}_{\text{freq}}$ and Stage 2 alignment. As shown in Table 7, removing $\mathcal{L}_{\text{freq}}$ decreases ACC by 2.45, 1.63, and 0.70 percentage points for

Table 6: *Effect of Stage 2 proxy proportion on NanoSet.*

Variant	CLIP ViT-B/16					SwinV2-Tiny					PGC				
	T-ACC↑	AUC↑	ACC↑	EER↓	FAR↓	T-ACC↑	AUC↑	ACC↑	EER↓	FAR↓	T-ACC↑	AUC↑	ACC↑	EER↓	FAR↓
Student	71.40	89.57	79.77	18.84	28.60	73.49	90.59	80.93	19.30	26.51	42.09	88.79	69.77	18.60	57.91
Mix	73.49	89.98	80.12	17.44	26.51	74.42	91.09	81.40	18.60	25.58	62.09	93.79	79.07	**14.19**	37.91
Proxy	**75.35**	**90.33**	**81.05**	**17.21**	**24.65**	**75.81**	**91.54**	**81.98**	**18.37**	**24.19**	**70.00**	**93.84**	**82.44**	14.42	**30.00**

Table 7: *Ablation of frequency loss and Stage 2 alignment on NanoSet.*

Variant	$\mathcal{L}_{\text{freq}}$	Stage 2	CLIP ViT-B/16					SwinV2-Tiny					Xception				
			T-ACC↑	AUC↑	ACC↑	EER↓	FAR↓	T-ACC↑	AUC↑	ACC↑	EER↓	FAR↓	T-ACC↑	AUC↑	ACC↑	EER↓	FAR↓
Variant 1	×	✓	70.00	90.24	78.60	17.67	30.00	**75.81**	90.45	80.35	19.07	**24.19**	92.56	88.81	83.37	19.30	7.44
Variant 2	✓	×	71.40	89.57	79.77	18.84	28.60	73.49	90.59	80.93	19.30	26.51	**94.88**	89.43	83.60	18.37	**5.12**
FPG (Ours)	✓	✓	**75.35**	**90.33**	**81.05**	**17.21**	**24.65**	**75.81**	**91.54**	**81.98**	**18.37**	**24.19**	93.49	**89.61**	**84.07**	**17.91**	6.51

CLIP ViT-B/16, SwinV2-Tiny, and Xception, respectively, and decreases T-ACC by 5.35 percentage points for CLIP ViT-B/16, indicating that frequency alignment yields more effective proxies. Removing Stage 2 while retaining $\mathcal{L}_{\text{freq}}$ decreases T-ACC by 3.95 and 2.32 percentage points on CLIP ViT-B/16 and SwinV2-Tiny, respectively, while EER worsens across all three detectors, demonstrating that detector-response alignment provides additional improvements in downstream detection.

5.6 Visualization Results

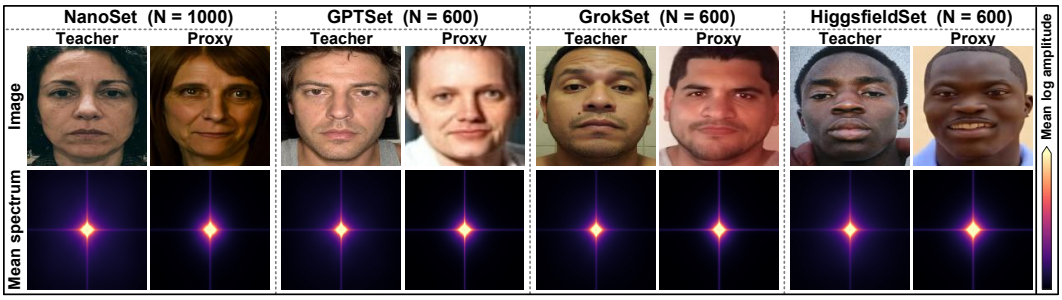

Figure 7: *Teacher-generated and proxy images (top) and average log-amplitude spectra (bottom).*

Fig. 7 shows that FPG proxies exhibit spectral structures similar to those of the corresponding teacher outputs across all four closed-source target-generator settings. Despite differences in image appearance, the proxy and teacher spectra remain closely aligned across the frequency domain, indicating that the proposed frequency alignment reduces their overall spectral discrepancy.

6 Conclusion

We formulated the new problem of adapting deepfake detectors to black-box generators under access and reuse constraints and introduced FPG to address it. FPG converts limited black-box supervision into reusable proxy images that transfer teacher-relevant generative and forensic characteristics for detector adaptation. Across six target generators, FPG improves detection of teacher-generated images and balanced real–fake data without directly using teacher outputs for detector training. Under limited teacher access, its proxies can outperform direct training on teacher outputs. Together, these results show the effectiveness of FPG for practical detector adaptation under constrained access.

Limitations. FPG depends on the capacity of the student generator, which affects how well teacher-specific generative and forensic characteristics are transferred. Stage 2 is also detector-dependent: Table 5 shows that detector-specific proxies outperform proxies guided by a shared detector. We also do not evaluate whether proxies closely replicate teacher outputs under very small query budgets.

Future work. Future work could improve forensic transfer across student architectures and develop detector-agnostic or multi-detector alignment for broader generalization. Extending FPG beyond facial imagery to other AI-generated image domains and modalities, such as video and audio, is another promising direction. Developing a unified single-stage framework is also worth exploring.

AI USE STATEMENT

In this work, we used generative AI tools to create and edit source code and to improve the readability and clarity of the paper. We reviewed all AI-assisted code and verified its correctness, and we reviewed all AI-assisted manuscript changes for consistency and accuracy. We take responsibility for the final content of this work, including all text and code produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

All source code necessary for our proposed framework is available in an anonymized codebase.

REFERENCES

Yoshua Bengio, Nicholas Léonard, and Aaron Courville. Estimating or propagating gradients through stochastic neurons for conditional computation, 2013. URL https://arxiv.org/abs/1308.3432.

Fadi Boutros, Jonas Henry Grebe, Arjan Kuijper, and Naser Damer. Idiff-face: Synthetic-based face recognition through fizzy identity-conditioned diffusion model. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 19650–19661, October 2023.

Francois Chollet. Xception: Deep learning with depthwise separable convolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

Yuxuan Duan, Yan Hong, Bo Zhang, Jun Lan, Huijia Zhu, Weiqiang Wang, Jianfu Zhang, Li Niu, and Liqing Zhang. Domaingallery: Few-shot domain-driven image generation by attribute-centric finetuning. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 537–561. Curran Associates, Inc., 2024. doi: 10.52202/079017-0017. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/013d743db3c684957305d32017f13339-Paper-Conference.pdf.

Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, Dustin Podell, Tim Dockhorn, Zion English, and Robin Rombach. Scaling rectified flow transformers for high-resolution image synthesis. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 12606–12633. PMLR, 2024. URL https://proceedings.mlr.press/v235/esser24a.html.

Joel Frank, Thorsten Eisenhofer, Lea Schönherr, Asja Fischer, Dorothea Kolossa, and Thorsten Holz. Leveraging frequency analysis for deep fake image recognition. In Hal Daumé III and Aarti Singh (eds.), *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 3247–3258. PMLR, 13–18 Jul 2020. URL https://proceedings.mlr.press/v119/frank20a.html.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K.Q. Weinberger (eds.), *Advances in Neural Information Processing Systems*, volume 27, pp. 2672–2680. Curran Associates, Inc., 2014.

Google. Gemini developer api pricing. https://ai.google.dev/gemini-api/docs/pricing, 2026a. Accessed September 26, 2026.

Google. Gemini api additional terms of service. https://ai.google.dev/gemini-api/terms, 2026b. Effective March 23, 2026; accessed September 26, 2026.

Google DeepMind. Introducing nano banana pro. https://blog.google/innovation-and-ai/products/nano-banana-pro/, November 2025. Published November 20, 2025; accessed September 26, 2026.

Higgsfield. Soul 2 api reference. https://open.higgsfield.ai/models/higgsfield-ai/soul/v2/standard/api-reference, 2026. Model ID: higgsfield-ai/soul/v2/standard; accessed September 26, 2026.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network, 2015. URL https://arxiv.org/abs/1503.02531.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 6840–6851. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper/2020/file/4c5bcfec8584af0d967f1ab10179ca4b-Paper.pdf.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=nZeVKeeFYf9.

Ziqi Huang, Kelvin C.K. Chan, Yuming Jiang, and Ziwei Liu. Collaborative diffusion for multi-modal face generation and editing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6080–6090, June 2023. URL https://openaccess.thecvf.com/content/CVPR2023/html/Huang_Collaborative_Diffusion_for_Multi-Modal_Face_Generation_and_Editing_CVPR_2023_paper.html.

Dimitrios Karageorgiou, Symeon Papadopoulos, Ioannis Kompatsiaris, and Efstratios Gavves. Any-resolution ai-generated image detection by spectral learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 18706–18717, June 2025.

Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. In *International Conference on Learning Representations*, 2014. URL http://arxiv.org/abs/1312.6114.

Lingzhi Li, Jianmin Bao, Ting Zhang, Hao Yang, Dong Chen, Fang Wen, and Baining Guo. Face x-ray for more general face forgery detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

Yiheng Li, Zichang Tan, Guoqing Xu, Zhen Lei, Xu Zhou, and Yang Yang. Towards generalizable ai-generated image detection via image-adaptive prompt learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 21262–21272, June 2026.

Shuqiao Liang, Jian Liu, Chen Renzhang, and Quanlong Guan. Ferretnet: Efficient synthetic image detection via local pixel dependencies. In D. Belgrave, C. Zhang, H. Lin, R. Pascanu, P. Koniusz, M. Ghassemi, and N. Chen (eds.), *Advances in Neural Information Processing Systems*, volume 38, Main Conference, pp. 42109–42133. Curran Associates, Inc., 2025. doi: 10.52202/085713-1402. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/3bfcc45018b4156b92197db47e9742bd-Paper-Conference.pdf.

Li Lin, Santosh Santosh, Mingyang Wu, Xin Wang, and Shu Hu. AI-Face: A million-scale demographically annotated AI-generated face dataset and fairness benchmark. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3503–3515, June 2025. URL https://openaccess.thecvf.com/content/CVPR2025/html/Lin_AI-Face_A_Million-Scale_Demographically_Annotated_AI-Generated_Face_Dataset_and_Fairness_CVPR_2025_paper.html.

Ze Liu, Han Hu, Yutong Lin, Zhuliang Yao, Zhenda Xie, Yixuan Wei, Jia Ning, Yue Cao, Zheng Zhang, Li Dong, Furu Wei, and Baining Guo. Swin transformer v2: Scaling up capacity and resolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 12009–12019, June 2022.

Mekhail Mustak, Joni Salminen, Matti Mäntymäki, Arafat Rahman, and Yogesh K. Dwivedi. Deepfakes: Deceptions, mitigations, and opportunities. *Journal of Business Research*, 154: 113368, 2023. ISSN 0148-2963. doi: 10.1016/j.jbusres.2022.113368. URL https://www.sciencedirect.com/science/article/pii/S0148296322008335.

Utkarsh Ojha, Yuheng Li, and Yong Jae Lee. Towards universal fake image detectors that generalize across generative models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 24480–24489, June 2023.

OpenAI. GPT Image 2. https://developers.openai.com/api/docs/models/gpt-image-2, 2026. Model ID: `gpt-image-2`; accessed September 26, 2026.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In Marina Meila and Tong Zhang (eds.), *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 8748–8763. PMLR, 18–24 Jul 2021. URL https://proceedings.mlr.press/v139/radford21a.html.

Naina Raisinghani. Nano banana is coming to google search, notebooklm and photos. https://blog.google/innovation-and-ai/products/nano-banana-google-products/, October 2025. Google DeepMind, accessed September 18, 2026.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10684–10695, June 2022. URL https://openaccess.thecvf.com/content/CVPR2022/html/Rombach_High-Resolution_Image_Synthesis_With_Latent_Diffusion_Models_CVPR_2022_paper.html.

Andreas Rossler, Davide Cozzolino, Luisa Verdoliva, Christian Riess, Justus Thies, and Matthias Niessner. FaceForensics++: Learning to detect manipulated facial images. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019. URL https://openaccess.thecvf.com/content_ICCV_2019/html/Rossler_FaceForensics_Learning_to_Detect_Manipulated_Facial_Images_ICCV_2019_paper.html.

Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Yael Pritch, Michael Rubinstein, and Kfir Aberman. Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 22500–22510, June 2023.

Kaede Shiohara and Toshihiko Yamasaki. Detecting deepfakes with self-blended images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 18720–18729, June 2022.

Anirudh Sundara Rajan and Yong Jae Lee. Stay-positive: A case for ignoring real image features in fake image detection. In Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu (eds.), *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 58008–58028. PMLR, 13–19 Jul 2025. URL https://proceedings.mlr.press/v267/sundara-rajan25a.html.

Mingxing Tan and Quoc Le. EfficientNet: Rethinking model scaling for convolutional neural networks. In Kamalika Chaudhuri and Ruslan Salakhutdinov (eds.), *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 6105–6114. PMLR, 09–15 Jun 2019. URL https://proceedings.mlr.press/v97/tan19a.html.

Hongsong Wang, Renxi Cheng, Yang Zhang, Chaolei Han, and Jie Gui. Lota: Bit-planes guided ai-generated image detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 17246–17255, October 2025.

xAI. Grok imagine quality mode api. https://x.ai/news/grok-imagine-quality-mode, May 2026. Published May 6, 2026; model ID: `grok-imagine-image-quality`; accessed September 26, 2026.

Zhiyuan Yan, Jiangming Wang, Peng Jin, Ke-Yue Zhang, Chengchun Liu, Shen Chen, Taiping Yao, Shouhong Ding, Baoyuan Wu, and Li Yuan. Orthogonal subspace decomposition for generalizable AI-generated image detection. In Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu (eds.), *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 70268–70288. PMLR, 13–19 Jul 2025. URL https://proceedings.mlr.press/v267/yan25b.html.

Yawen Yang, Feng Li, Shuqi Kong, Yunfeng Diao, Xinjian Gao, Zenglin Shi, and Meng Wang. Layer consistency matters: Elegant latent transition discrepancy for generalizable synthetic image detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 38111–38121, June 2026.

Tianwei Yin, Michaël Gharbi, Taesung Park, Richard Zhang, Eli Shechtman, Frédo Durand, and William T. Freeman. Improved distribution matching distillation for fast image synthesis. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 47455–47487. Curran Associates, Inc., 2024a. doi: 10.52202/079017-1505. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/54dcf25318f9de5a7a01f0a4125c541e-Paper-Conference.pdf.

Tianwei Yin, Michaël Gharbi, Richard Zhang, Eli Shechtman, Frédo Durand, William T. Freeman, and Taesung Park. One-step diffusion with distribution matching distillation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6613–6623, June 2024b.

Lin Yuan, Xiaowan Li, Yan Zhang, Jiawei Zhang, Hongbo Li, and Xinbo Gao. Mlep: Multi-granularity local entropy patterns for generalized ai-generated image detection. In D. Belgrave, C. Zhang, H. Lin, R. Pascanu, P. Koniusz, M. Ghassemi, and N. Chen (eds.), *Advances in Neural Information Processing Systems*, volume 38, Main Conference, pp. 68981–69000. Curran Associates, Inc., 2025. doi: 10.52202/085713-2321. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/63db94e06ce85dde10871c99dd39ad98-Paper-Conference.pdf.

Sicheng Zhao, Guangzhi Wang, Shanghang Zhang, Yang Gu, Yaxian Li, Zhichao Song, Pengfei Xu, Runbo Hu, Hua Chai, and Kurt Keutzer. Multi-source distilling domain adaptation. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(07):12975–12983, 2020. doi: 10.1609/aaai.v34i07.6997. URL https://ojs.aaai.org/index.php/AAAI/article/view/6997.

Yinglin Zheng, Hao Yang, Ting Zhang, Jianmin Bao, Dongdong Chen, Yangyu Huang, Lu Yuan, Dong Chen, Ming Zeng, and Fang Wen. General facial representation learning in a visual-linguistic manner. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 18697–18709, June 2022.

Xiaoyu Zhou, Jianwei Fei, Peipeng Yu, Jingchang Xie, Chong Cheng, and Zhihua Xia. PGC: Peak-guided calibration for generalizable AI-generated image detection. In *Proceedings of the 43rd International Conference on Machine Learning*, 2026.

A Appendix

A.1 Additional Method Details

A.1.1 Black-Box Adaptation of DMD2

DMD2 (Yin et al., 2024a) is a distribution-matching distillation method that trains a student generator from the difference between target and fake scores. In the original formulation, the target score comes from a teacher diffusion model, while the fake score is estimated by a diffusion model trained on student-generated images. Their difference provides the gradient for student optimization.

For the black-box frequency study in Sec. 3, this formulation is not directly applicable because the target generator exposes only generated images rather than its diffusion model or internal score. We use a black-box adaptation of DMD2 in which collected teacher-generated images provide the target-side supervision required for distribution matching, while student-generated images are kept on the fake side. This allows the student to be optimized from only the teacher's output distribution.

A.1.2 Teacher-Guided Frequency Alignment

For frequency alignment, each decoded RGB image x is converted to luminance via the mapping $Y(x) = 0.299x_R + 0.587x_G + 0.114x_B$. We then compute its orthonormally normalized 2-D Fourier transform and define the resulting log-amplitude spectrum as $\mathcal{F}(x) = \log\left(1 + |\mathrm{FFT2}_{\mathrm{ortho}}(Y(x))|\right)$. For an image of size $H \times W$, we exclude the zero-frequency (DC) coefficient and compute the mean squared error over the remaining $HW - 1$ frequency components:

$$\mathcal{L}_{\mathrm{freq}} = \mathbb{E}_{(x_i, c_i, x_i^T) \sim \mathcal{I}_T} \frac{1}{HW - 1} \sum_{(k,\ell) \neq (0,0)} \left(\mathcal{F}(x_i^S)_{k,\ell} - \mathcal{F}(x_i^T)_{k,\ell}\right)^2 .$$

The loss is evaluated on continuous-valued decoded images before quantization.

A.1.3 Stage 1 with Flow Matching

For experiments using SD3 (Esser et al., 2024), we replace the diffusion reconstruction loss with its rectified-flow objective. For $(x, c) \in \mathcal{P}$, we map x to a latent $z_0 = E(x)$ using the frozen VAE, sample Gaussian noise $\epsilon \sim \mathcal{N}(0, I)$ and a noise level σ_t, and construct $z_t = (1 - \sigma_t)z_0 + \sigma_t\epsilon$. Given z_t, the student predicts the flow $v_S(z_t, t, c)$ toward the target $\epsilon - z_0$. The flow-matching loss is

$$\mathcal{L}_{\mathrm{flow}} = \mathbb{E}_{(x,c) \sim \mathcal{P},\, t,\, \epsilon} \left\| v_S(z_t, t, c) - (\epsilon - z_0) \right\|_2^2 .$$

For teacher-guided latent alignment, we recover the predicted clean latent as $\hat{z}_0^S = z_t - \sigma_t v_S(z_t, t, c)$ and use it to compute $\mathcal{L}_{\mathrm{latent}}$ as defined in Sec. 4.2. The frequency alignment is unchanged: student-generated images are compared with the corresponding teacher-generated images using $\mathcal{L}_{\mathrm{freq}}$.

A.1.4 Quantization-Aware Detector-Response Alignment

Stage 2 optimizes the perturbation δ continuously, while the resulting proxy images are stored as 8-bit images. To account for this mismatch, each forward pass evaluates the quantized image $\tilde{x}^S = Q_8(\mathrm{clip}(x^S + \delta))$, where Q_8 rounds pixel values to the nearest 8-bit level. Because this operation is non-differentiable, we use a straight-through estimator (STE) (Bengio et al., 2013) so that the forward pass uses the quantized image while gradients are propagated through the continuous image. The alignment objective therefore becomes

$$\mathcal{L}_{\mathrm{align}}(\delta) = \frac{1}{2}\|\delta\|_2^2 + \lambda \left| p_D(\tilde{x}^S) - p_D(x^T) \right| .$$

A.2 Additional Evaluation Details

Relationship Between T-ACC and FAR. T-ACC measures accuracy on teacher-generated fake images, while FAR measures the fraction of these images classified as real under the same threshold. Thus, in our evaluation setting, the two metrics are complementary, with $\mathrm{FAR} = 100\% - \mathrm{T\text{-}ACC}$. We report both for consistency across the teacher-only and mixed evaluation settings.

A.3 Additional Experimental Results

A.3.1 Computational Cost

Table 8: *Runtime of the NanoSet SD3 Medium–Xception pipeline on a single NVIDIA RTX A6000.*

Stage	Timed workload	Total time	Normalized time
Stage 1 training	10,000 updates	6.54 h	2.356 s/update
Candidate generation	8 candidates	31.99 ± 0.23 s	3.999 ± 0.029 s/image
Stage 2 alignment	64 pairs	34.47 ± 1.63 s	538.64 ± 25.41 ms/image
Final proxy generation	8 proxies	49.70 ± 0.14 s	6.212 ± 0.017 s/image

Table 8 reports the runtime of the NanoSet SD3 Medium–Xception pipeline on a single NVIDIA RTX A6000. Timings are reported as the mean $\pm$ sample standard deviation over three repetitions after warmup. Stage 1 training requires 6.54 hours for 10,000 updates, corresponding to 2.36 seconds per update. After training, candidate generation takes approximately 4.00 seconds per image, while 40 Stage 2 alignment steps add approximately 0.54 seconds per image at a batch size of two. Producing a final proxy image from an input prompt takes 6.21 seconds on average at a batch size of one, including generation, face cropping, alignment, and image saving.

A.3.2 Robustness to Image Transformations

Images may undergo compression, resizing, or blurring during storage, transmission, and preprocessing, potentially altering features used for detection. We therefore evaluate frozen Xception detectors trained with FPG proxies on GPTSet, CollabSet, and IDiffSet under JPEG compression, resizing, and Gaussian blur. Each transformation is applied independently to both real and teacher-generated evaluation images, while model parameters and classification thresholds remain fixed.

As shown in Table 9, detection of teacher outputs remains robust to resizing and Gaussian blur, with T-ACC unchanged or improved across all three datasets. JPEG compression causes comparatively smaller performance changes. In contrast, resizing and blur reduce AUC and ACC, particularly on CollabSet and IDiffSet, despite maintaining high T-ACC. This indicates that these transformations degrade discrimination of the real images rather than detection of teacher-generated images.

Table 9: *Effect of common image transformations on frozen detectors trained with FPG proxies. Real and fake images receive the same transformation: JPEG compression at quality 75, bicubic down/up resizing by a factor of 0.5, or Gaussian blur with a 7×7 kernel and $\sigma = 1.0$.*

Transformation	**GPTSet**†					**CollabSet***					**IDiffSet***				
	T-ACC↑	AUC↑	ACC↑	EER↓	FAR↓	T-ACC↑	AUC↑	ACC↑	EER↓	FAR↓	T-ACC↑	AUC↑	ACC↑	EER↓	FAR↓
JPEG ($Q = 75$)	91.25	91.71	85.25	16.25	8.75	97.90	93.86	81.66	13.52	2.10	99.68	95.68	83.93	10.99	0.32
Resize ($0.5\times$)	96.00	91.48	83.88	16.25	4.00	98.64	90.67	79.45	18.13	1.36	**100.00**	92.83	75.69	14.38	**0.00**
Gaussian blur ($\sigma = 1.0$)	**98.50**	91.42	80.13	14.75	**1.50**	**98.74**	88.10	71.33	20.34	**1.26**	**100.00**	90.23	69.24	17.86	**0.00**
FPG (Clean)	93.75	**92.51**	**85.88**	**14.50**	6.25	98.22	**94.96**	**87.68**	**12.79**	1.78	**100.00**	**96.69**	**85.68**	**8.56**	**0.00**

A.3.3 Stage 2 Response Targets

We examine how the response target affects Stage 2 proxy generation. We replace the detector response of the paired teacher-generated image with either the mean response over teacher-generated images or a fixed fake probability of 1.0, while keeping the remaining protocol unchanged. As shown in Table 10, using the mean response produces only small, mixed changes: AUC differs by at most 0.06 percentage points, while T-ACC and FAR remain unchanged. In contrast, the fixed target performs worse across all reported metrics and datasets. For example, AUC decreases from 89.61% to 87.37% on NanoSet and from 97.75% to 96.16% on GrokSet. These results support using teacher-response guidance rather than a fixed fake target for Stage 2 alignment.

A.3.4 Stage 2 Iteration Budget

We examine whether Stage 2 requires a large optimization budget by varying the iteration count from 10 to 80 on NanoSet, GrokSet, and GPTSet while keeping all other settings fixed. As shown in Table 11, performance varies little across the four budgets. Relative to the default 40 steps, AUC

Table 10: *Effect of the Stage 2 response target on downstream Xception performance.*

Target	NanoSet					GrokSet					GPTSet				
	T-ACC↑	AUC↑	ACC↑	EER↓	FAR↓	T-ACC↑	AUC↑	ACC↑	EER↓	FAR↓	T-ACC↑	AUC↑	ACC↑	EER↓	FAR↓
Mean teacher	**93.49**	89.55	**84.19**	18.37	**6.51**	**94.75**	**97.78**	91.88	7.75	**5.25**	**93.75**	**92.51**	**86.00**	14.75	**6.25**
Fixed fake 1.0	88.60	87.37	82.21	19.77	11.40	88.75	96.16	89.50	10.50	11.25	90.00	91.83	84.75	16.00	10.00
FPG	**93.49**	**89.61**	84.07	**17.91**	**6.51**	**94.75**	97.75	**92.00**	**7.50**	**5.25**	**93.75**	**92.51**	85.88	**14.50**	**6.25**

Table 11: *Effect of the Stage 2 iteration budget on Xception performance.*

Steps	NanoSet					GrokSet					GPTSet				
	T-ACC↑	AUC↑	ACC↑	EER↓	FAR↓	T-ACC↑	AUC↑	ACC↑	EER↓	FAR↓	T-ACC↑	AUC↑	ACC↑	EER↓	FAR↓
10	**93.49**	89.61	**84.19**	17.91	**6.51**	**94.75**	97.74	91.88	**7.50**	**5.25**	**93.75**	92.49	**85.88**	**14.50**	**6.25**
20	**93.49**	**89.66**	**84.19**	**17.67**	**6.51**	**94.75**	**97.77**	**92.00**	**7.50**	**5.25**	**93.75**	92.49	**85.88**	**14.50**	**6.25**
40 (FPG)	**93.49**	89.61	84.07	17.91	**6.51**	**94.75**	97.75	**92.00**	**7.50**	**5.25**	**93.75**	92.51	**85.88**	**14.50**	**6.25**
80	**93.49**	89.64	84.07	17.91	**6.51**	**94.75**	97.75	**92.00**	7.75	**5.25**	**93.75**	**92.54**	**85.88**	**14.50**	**6.25**

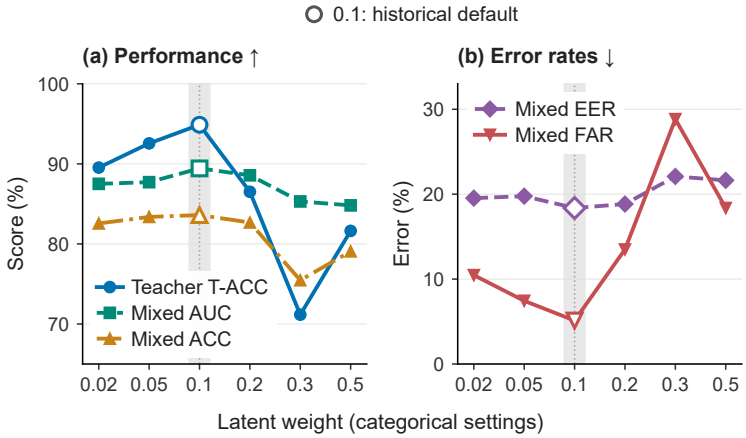

Figure 8: *Effect of the Stage 1 latent weight on downstream Xception performance on NanoSet.*

differs by at most 0.05 percentage points, while T-ACC and FAR remain unchanged. For example, GPTSet AUC increases only from 92.51% at 40 steps to 92.54% at 80 steps. These results indicate that downstream detection is largely insensitive to the Stage 2 iteration budget over the tested range.

A.3.5 Stage 1 Latent-Weight Sensitivity

To assess how the Stage 1 latent weight affects downstream detection, we vary λ_{latent} in the default Stage 1 configuration on NanoSet and evaluate Xception trained on resulting student-generated images without Stage 2 alignment. For each setting, we recalibrate λ_{freq} at the start of training and keep it fixed thereafter. As shown in Fig. 8, 0.1 performs best across reported metrics, achieving 89.43% AUC and 83.60% ACC. Nearby settings remain competitive: 0.05 reaches 83.37% ACC, while 0.2 reaches 88.57% AUC. Larger weights yield weaker performance, with AUC dropping to 85.31% and 84.81% at 0.3 and 0.5, respectively. These results support 0.1 as the default latent weight.

A.4 Dataset Details

We evaluate FPG on six datasets spanning four closed-source and two open-source generators. The closed-source settings use AI-Face as the permissible data for detector training, while the open-source settings use FaceForensics++. We follow the dataset splits described in Sec. 5.1. Teacher-generated images guide proxy generation but are not used directly to train downstream detectors.

For GPTSet, GrokSet, and HiggsfieldSet, we report the generation tools, exact model identifiers, and collection dates. GPTSet uses OpenAI GPT Image 2 at medium quality, identified as `gpt-image-2`, with two collection batches completed on June 5 and June 11, 2026. GrokSet uses xAI Grok Imagine in image quality mode, identified as `grok-imagine-image-quality`, with

images collected from May 30 to June 4, 2026. HiggsfieldSet uses Higgsfield Soul v2 Standard at a 9:16 aspect ratio and 1080p output, identified as `higgsfield-ai/soul/v2/standard`, with images collected from April 24 to June 4, 2026. Each dataset uses 600 teacher-generated images for proxy generation and 400 disjoint teacher-generated images for evaluation. NanoSet uses Nano Banana Pro, with 1,000 teacher-generated images for proxy generation and 430 disjoint images for evaluation, collected from February 10 to February 13, 2026.

Figures 9 and 10 compare teacher-generated images with final FPG proxies across all six datasets. Each dataset occupies two rows, with teacher outputs above and proxies below; each row contains six RGB examples and a mean spectrum. Because FPG does not reconstruct individual teacher-generated images, facial appearance, expressions, and local details may differ. Mean spectra are computed separately over teacher-generated and proxy images. Despite these content differences, the two exhibit similar overall spectral structures. These visualizations extend the qualitative analysis in Sec. 5.6 and provide additional evidence of spectral similarity in FPG proxies.

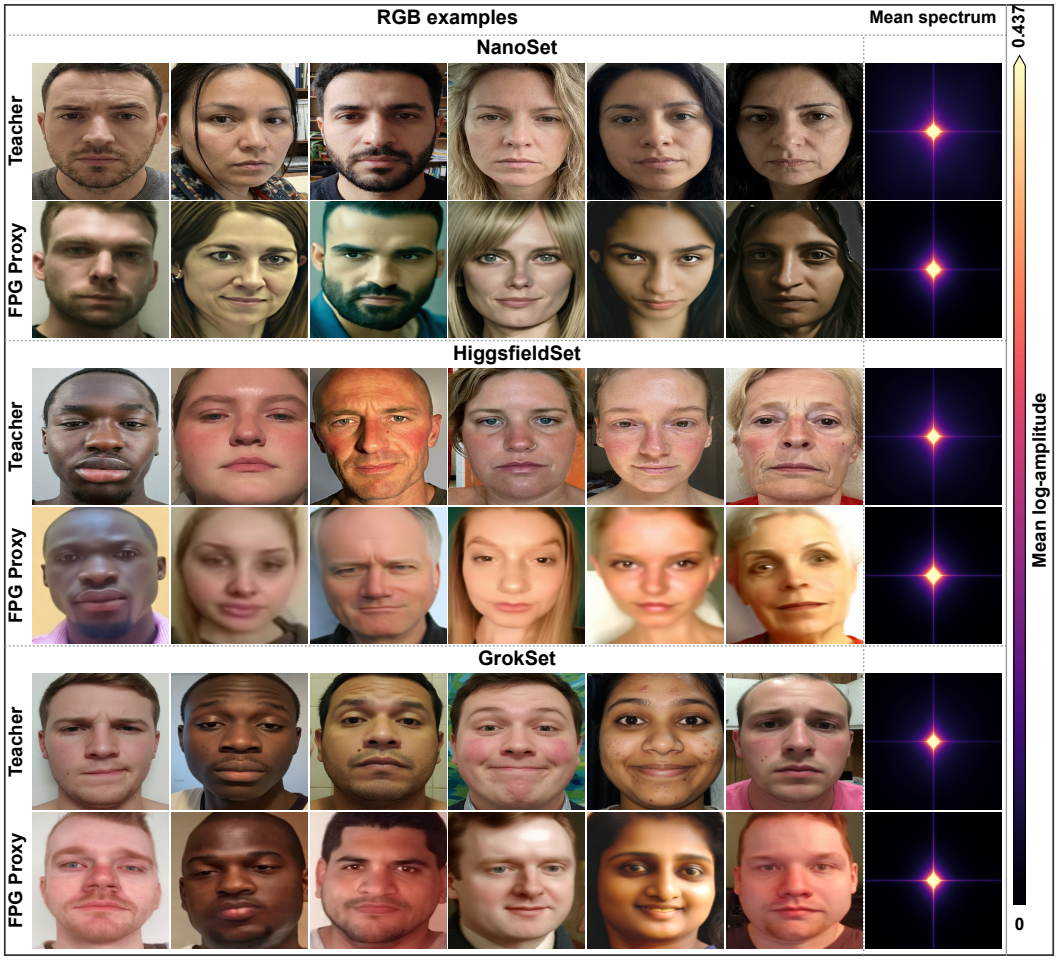

Figure 9: *Teacher-generated and FPG proxy examples for NanoSet, HiggsfieldSet, and GrokSet. Each dataset shows six RGB examples and the corresponding mean spectrum, with teacher-generated images above and FPG proxies below.*

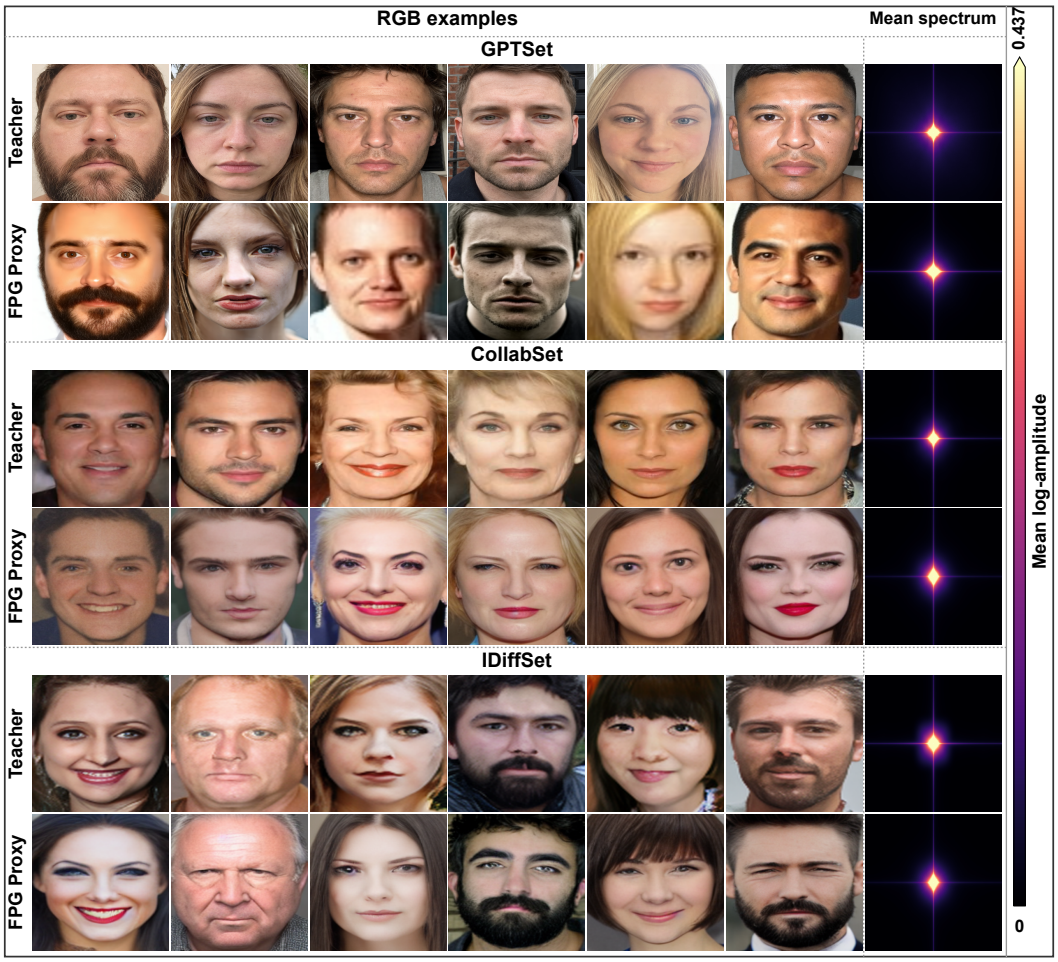

Figure 10: *Teacher-generated and FPG proxy examples for GPTSet, CollabSet, and IDiffSet. Each dataset shows six RGB examples and the corresponding mean spectrum, with teacher-generated images above and FPG proxies below.*