GENERATIVE CLASSIFICATION VIA FLOW MATCHING FOR AI-GENERATED IMAGE DETECTION

Anonymous authors
Paper under double-blind review

ABSTRACT

The rapid evolution of generative models and the growing diversity of generation sources pose persistent challenges to the cross-generator generalization of AI-generated image detectors. Existing approaches primarily improve detection by exploiting forensic artifacts, enhancing visual representations, or adapting pretrained models. However, they typically formulate authenticity prediction as a direct mapping, which may encourage reliance on salient cues specific to training generators. To address this limitation, we propose ***FakeFlow***, a generative classification framework based on Flow Matching that formulates authenticity prediction as an image-conditioned continuous process in label space. Given representations from a frozen visual encoder, FakeFlow learns conditional trajectories from Gaussian initial states toward real or fake label vectors. We instantiate FakeFlow with two complementary parameterizations: ***FakeFlow***$_{Drift}$ regresses the trajectory velocity field, whereas ***FakeFlow***$_{Leap}$ predicts the target label vector and converts it into the corresponding update velocity. Although both formulations admit an equivalent one-step decision rule under ideal conditions, their different temporal error weighting leads to complementary generalization behavior. We further incorporate Classifier-Free Guidance (CFG) to amplify the discrepancy between image-conditioned and unconditional predictions, encouraging the final decision to rely more strongly on image-specific evidence. Experiments on GenImage, AIGI-Holmes, AIGI-Now, and In-the-Wild demonstrate strong generalization across unseen generators and real-world settings. Code is available at https://anonymous.4open.science/r/fakeflow-7575.

1 INTRODUCTION

The rapid advancement of generative models has substantially improved the visual quality of synthetic images, while posing growing challenges to visual content authenticity and digital forensics. As generation sources continue to diversify, new architectures, model variants, and creation tools introduce evolving forgery patterns, whereas detector training typically covers only a limited set of sources. Consequently, a key challenge in AI-generated image detection is to generalize beyond known sources to unseen generators and complex real-world scenarios.

Existing methods primarily improve cross-generator detection by enhancing *visual evidence*. One line of work exploits low-level forensic traces left by the generation process, including frequency anomalies, texture patterns, and local pixel correlations (Tan et al., 2024a; Liu et al., 2020; Tan et al., 2024b). Another line leverages general representations from visual foundation models and improves cross-generator transfer through linear classification, feature adaptation, or semantic–artifact disentanglement (Ojha et al., 2023; Yan et al., 2025; Guo et al., 2026). Moreover, AIDE combines high-level semantic features with low-level noise patterns (Yan et al., 2024), while PPL employs patch-level learning to reduce reliance on a few local regions (Yang et al., 2026). Despite their methodological differences, these approaches primarily focus on extracting or organizing visual evidence that generalizes across generators.

However, forgery traces produced by unseen generators may differ substantially from those observed during training, making decision rules built on source-specific cues difficult to generalize. Prior studies have shown that detectors tend to overfit to a limited set of forgery cues or rely excessively on locally salient artifacts (Yan et al., 2025; Yang et al., 2026). We argue that this issue concerns

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.

not only *what visual representations are learned*, but also *how visual evidence is translated into the final decision*. In conventional classification, a model predicts the class through a direct mapping. Once a few salient cues suffice to separate the training samples, the classification objective can be satisfied without requiring the model to exploit additional image evidence.

Motivated by this observation, we adopt a different classification paradigm: rather than directly mapping an image to a class, the model starts from an uncertain state and progressively forms its decision under image guidance. During training, each image is associated with intermediate states of varying uncertainty, requiring the model to predict the appropriate evolution at different points along the trajectory. Consequently, classification is no longer supervised only at a single endpoint, but throughout the entire label trajectory.

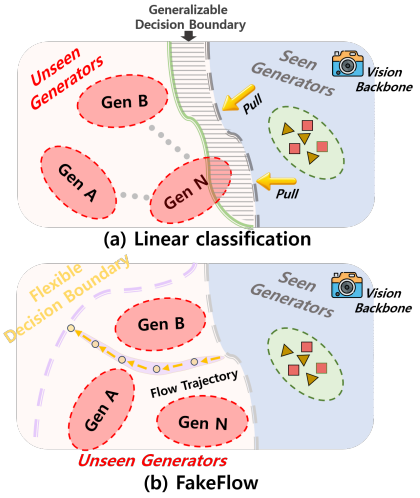

Figure 1: Linear classification vs. FakeFlow. FakeFlow predicts authenticity through image-conditioned trajectories in label space.

Flow Matching provides a natural framework for this paradigm by modeling the transport from an initial distribution to a target distribution through continuous states and a velocity field. Rather than modeling the generative process in image space, we perform this transport in a low-dimensional label space, conditioned on representations from a frozen visual encoder. Building on this formulation, we propose **FakeFlow**, a generative classification framework that models authenticity prediction as an image-guided continuous prediction process in label space.

Within the same label flow, we investigate two parameterizations. **FakeFlow**$_{\text{Drift}}$ directly predicts the trajectory velocity field, specifying the direction and magnitude of state updates, whereas **FakeFlow**$_{\text{Leap}}$ predicts the target label at the trajectory endpoint and converts it into the corresponding update velocity. Under optimal conditions, the two parameterizations yield the same one-step decision but induce different temporal error weighting during training. This allows us to study how parameterization affects generalization while preserving the same target decision rule.

For unseen generators, forgery traces may deviate from the statistics observed during training, making it important for the final decision to rely sufficiently on evidence from the input image. We therefore introduce Classifier-Free Guidance (CFG) to amplify the difference between image-conditioned and unconditional predictions, thereby strengthening the influence of image-specific evidence on the label trajectory and final decision.

We evaluate FakeFlow on four benchmarks: GenImage, AIGI-Holmes, AIGI-Now, and In-the-Wild. Compared with the best existing specialized detector on each benchmark, FakeFlow$_{\text{Drift}}$ and FakeFlow$_{\text{Leap}}$ improve average accuracy by approximately 8.1 and 8.5 percentage points, respectively. The two parameterizations exhibit complementary strengths: FakeFlow$_{\text{Leap}}$ achieves the highest average accuracy across all three generator-based benchmarks, whereas FakeFlow$_{\text{Drift}}$ performs better on real-world web images. Our main contributions are summarized as follows:

- We study AI-generated image detection from the perspective of decision-process modeling, formulating authenticity prediction as an image-conditioned continuous process in label space. This provides a complementary perspective to existing representation-centric approaches.
- We propose FakeFlow with two complementary parameterizations: FakeFlow$_{\text{Drift}}$ learns the velocity field of label trajectories, while FakeFlow$_{\text{Leap}}$ directly predicts their target labels. We further incorporate CFG to strengthen image-conditioned evidence and enable efficient one-step inference.
- Across four cross-generator and real-world benchmarks, FakeFlow$_{\text{Drift}}$ and FakeFlow$_{\text{Leap}}$ improve average accuracy over the best existing specialized detectors by approximately 8.1 and 8.5 percentage points, respectively, demonstrating strong generalization to unseen generators.

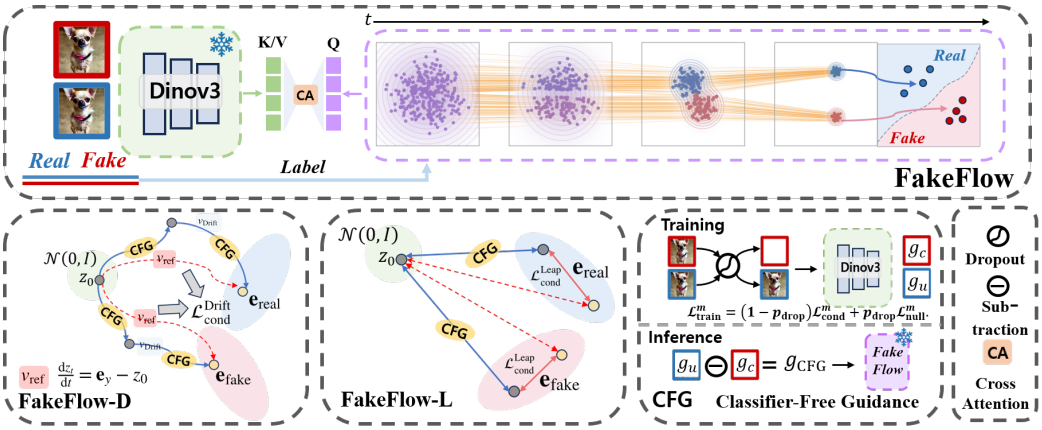

Figure 2: Overview of FakeFlow. A frozen DINOv3 encoder provides image conditions for label-space flow. $\text{FakeFlow}_{\text{Drift}}$ predicts trajectory velocities, whereas $\text{FakeFlow}_{\text{Leap}}$ predicts target labels. Classifier-Free Guidance (CFG) amplifies the influence of image evidence on the final prediction.

2 Related Work

Existing methods for generalizable AI-generated image detection mainly follow two directions. Forensic-based approaches exploit generation traces such as fingerprints, texture statistics, frequency anomalies, and local pixel correlations (Wang et al., 2020; Liu et al., 2020; Tan et al., 2024a;b). Subsequent methods further improve generalization through image transformations, feature disentanglement, data alignment, prior debiasing, and complementary forensic cues (Li et al., 2025a; Yan et al., 2025; Chen et al., 2025b; Zhou et al., 2025a; Wang et al., 2026; Dhakal et al., 2026). Another line of work leverages visual foundation models: UnivFD trains a lightweight classifier on frozen CLIP features (Ojha et al., 2023; Radford et al., 2021), while OmniAID and CLIP-ADA learn more transferable artifact representations (Guo et al., 2026; Deng et al., 2026). Recent studies further show that frozen features from modern foundation models such as DINOv3 already provide strong separability between real and generated images (Siméoni et al., 2025; Zhou et al., 2026).

3 Preliminaries

We briefly review Flow Matching and Classifier-Free Guidance (CFG), which form the basis of our label-space classification framework.

Flow Matching. Flow Matching learns a time-dependent vector field that transports samples from a source distribution p_0 to a target distribution p_1 (Lipman et al., 2023). Given independently sampled $z_0 \sim p_0$ and $z_1 \sim p_1$, we consider the linear probability path

$$z_t = (1-t)z_0 + tz_1, \qquad t \in [0,1], \tag{1}$$

whose conditional velocity is $z_1 - z_0$. A parameterized vector field $v_\theta(z_t, t)$ is trained by

$$\mathcal{L}_{\text{FM}}(\theta) = \mathbb{E}_{t,z_0,z_1}\left[\frac{1}{2}\left\|v_\theta(z_t,t) - (z_1 - z_0)\right\|_2^2\right], \tag{2}$$

where $t \sim \pi(t)$, typically with $\pi(t) = \mathcal{U}(0,1)$. The learned vector field defines the ODE $\mathrm{d}z/\mathrm{d}t = v_\theta(z,t)$, whose integration transports samples along the learned flow. For conditional Flow Matching, an external condition c is introduced as $v_\theta(z, t \mid c)$.

Classifier-Free Guidance. CFG jointly learns conditional and unconditional predictions by randomly dropping the condition during training (Ho & Salimans, 2022). For flow models, the guided velocity is

$$u_\theta(z_t, t \mid c) = (1-\omega)v_\theta(z_t, t \mid \varnothing) + \omega v_\theta(z_t, t \mid c), \tag{3}$$

where $\varnothing$ denotes the null condition and ω is the guidance scale. When $\omega = 1$, the conditional prediction is recovered; when $\omega > 1$, the prediction is extrapolated along the difference between the conditional and unconditional predictions.

4 Proposed Method

We formulate AI-generated image detection as an image-conditioned transport process in label space, constructing a continuous path from a Gaussian initial state to an authenticity label under visual conditioning. The time variable t parameterizes the interpolation between the initial state and the target label. Within this framework, we introduce two complementary parameterizations, **FakeFlow**$_{\mathrm{Drift}}$ and **FakeFlow**$_{\mathrm{Leap}}$, which share the same label space, linear path, and network architecture, but differ in their prediction targets and temporal weighting of training errors.

Our method consists of three components: **(a) Velocity Field and Target Prediction** (Sec. 4.1); **(b) Image-Guided Trajectory Prediction** (Sec. 4.2); and **(c) Training and Inference Strategy** (Sec. 4.3).

4.1 Velocity Field and Target Prediction

Let $m \in \{\mathrm{Drift}, \mathrm{Leap}\}$ denote the parameterization and $\mathcal{F}_{\theta_m}$ the corresponding prediction network, whose parameters are learned separately. All objectives are defined over $(I, y) \sim \mathcal{D}$, independently sampled Gaussian initial states, and $t \sim \pi(t)$.

From Noise to Label: Continuous Path Construction. Given an image-label pair $(I, y) \sim \mathcal{D}$, we represent its class as $z_1 = \mathbf{e}_y \in \mathbb{R}^C$, where $C = 2$, $y \in \{1, \ldots, C\}$, and $\mathbf{e}_y$ is the standard basis vector for class y. We independently sample $z_0 \sim \mathcal{N}(0, \mathbf{I}_C)$ and substitute $z_1 = \mathbf{e}_y$ into Eq. equation 1 to obtain z_t. The time variable t controls the interpolation between the initial state and the target label.

Drift Flow Modeling (FakeFlow$_{\mathrm{Drift}}$). The derivative of the linear path gives the target velocity $\mathbf{e}_y - z_0$. FakeFlow$_{\mathrm{Drift}}$ directly predicts this velocity:

$$v_{\mathrm{Drift}}(z_t, t \mid I) = \mathcal{F}_{\theta_{\mathrm{Drift}}}(z_t, t \mid I). \tag{4}$$

Its conditional training objective is

$$\mathcal{L}_{\mathrm{cond}}^{\mathrm{Drift}}(\theta_{\mathrm{Drift}}) = \mathbb{E}_{I,y,z_0,t}\left[\frac{1}{2}\left\|\mathcal{F}_{\theta_{\mathrm{Drift}}}(z_t, t \mid I) - (\mathbf{e}_y - z_0)\right\|_2^2\right]. \tag{5}$$

For a given initial state and target label, the target velocity remains constant along the linear path. FakeFlow$_{\mathrm{Drift}}$ therefore learns the image-conditioned velocity field and obtains the final prediction through state updates.

Leap Flow Modeling (FakeFlow$_{\mathrm{Leap}}$). FakeFlow$_{\mathrm{Leap}}$ instead directly predicts the target label vector $\mathbf{e}_y$ (Li & He, 2026). The predicted target is converted into a velocity as

$$v_{\mathrm{Leap}}(z_t, t \mid I) = \frac{\mathcal{F}_{\theta_{\mathrm{Leap}}}(z_t, t \mid I) - z_t}{r(t)}, \qquad r(t) = \max(1 - t, \epsilon), \tag{6}$$

where $0 < \epsilon \leq 1$ stabilizes the scaling near the trajectory endpoint. Applying the same transformation to the target gives

$$\begin{aligned}\mathcal{L}_{\mathrm{cond}}^{\mathrm{Leap}}(\theta_{\mathrm{Leap}}) &= \mathbb{E}_{I,y,z_0,t}\left[\frac{1}{2}\left\|\frac{\mathcal{F}_{\theta_{\mathrm{Leap}}}(z_t, t \mid I) - z_t}{r(t)} - \frac{\mathbf{e}_y - z_t}{r(t)}\right\|_2^2\right] \\ &= \mathbb{E}_{I,y,z_0,t}\left[\frac{\left\|\mathcal{F}_{\theta_{\mathrm{Leap}}}(z_t, t \mid I) - \mathbf{e}_y\right\|_2^2}{2r(t)^2}\right].\end{aligned} \tag{7}$$

Thus, FakeFlow$_{\mathrm{Leap}}$ is equivalent to time-weighted label regression, where $r(t)^{-2}$ determines the error weight at each time point. When $1 - t \geq \epsilon$, the target velocity reduces to $(\mathbf{e}_y - z_t)/(1 - t) = \mathbf{e}_y - z_0$.

Discrete Decoding. After state evolution, both parameterizations produce a terminal state $z_m^{(S)}$. We decode its class by nearest-neighbor matching to the basis vectors. Since $\|z_m^{(S)} - \mathbf{e}_j\|_2^2 = \|z_m^{(S)}\|_2^2 + 1 - 2[z_m^{(S)}]_j$,

$$\arg\min_{j\in\{1,\dots,C\}} \|z_m^{(S)} - \mathbf{e}_j\|_2^2 = \arg\max_{j\in\{1,\dots,C\}} [z_m^{(S)}]_j. \tag{8}$$

Thus, the predicted class is simply given by the largest component of the terminal state, without requiring an additional decoder or probability normalization.

4.2 Image-Guided Trajectory Prediction

We use cross-attention to extract visual evidence conditioned on the current state and time. A frozen visual encoder $\mathcal{E}$ extracts image tokens, while the state and time are jointly encoded as a query:

$$H_m = \mathcal{P}_{\theta_m}\big(\mathcal{E}(I)\big), \qquad q_t^m = \mathcal{S}_{\theta_m}(z_t) + \mathcal{T}_{\theta_m}(t). \tag{9}$$

Here, $H_m \in \mathbb{R}^{N\times d}$ denotes the projected visual tokens and $q_t^m \in \mathbb{R}^d$ the state–time query. $\mathcal{S}_{\theta_m}$ is an MLP state encoder, and $\mathcal{T}_{\theta_m}$ combines sinusoidal time encoding with an MLP.

Using q_t^m as the query and H_m as keys and values, we define

$$\mathcal{F}_{\theta_m}(z_t, t \mid I) = W_o^m \,\mathrm{LN}_m\left[q_t^m + \mathrm{MHA}_m(q_t^m, H_m, H_m)\right] + b_o^m. \tag{10}$$

The output represents a velocity for FakeFlow$_{\mathrm{Drift}}$ and a target-label estimate for FakeFlow$_{\mathrm{Leap}}$. The visual encoder $\mathcal{E}$ remains frozen, while the projection, state and time encoders, cross-attention module, and output layer are trainable.

4.3 Training and Inference Strategy

Training with Conditional Dropout. For each (I, y), we sample z_0 and $t \sim \pi(t)$ and construct z_t. We use $\pi(t) = \mathcal{U}(0,1)$ by default and study non-uniform sampling with a Kumaraswamy distribution parameterized by (α, β), where $\alpha = \beta = 1$ recovers uniform sampling. For FakeFlow$_{\mathrm{Leap}}$, the sampled time additionally affects the loss through the weighting $r(t)^{-2}$ in Eq. equation 7.

To enable CFG, we independently drop the image condition with probability p_{drop} while retaining the same state and supervision target. The null condition $\varnothing$ is implemented by replacing the normalized image with an all-zero input, processed by the same visual encoder and projection module. The overall objective is

$$\mathcal{L}_{\mathrm{train}}^m(\theta_m) = (1 - p_{\mathrm{drop}})\mathcal{L}_{\mathrm{cond}}^m(\theta_m) + p_{\mathrm{drop}}\mathcal{L}_{\mathrm{null}}^m(\theta_m), \qquad m \in \{\mathrm{Drift}, \mathrm{Leap}\}, \tag{11}$$

where $\mathcal{L}_{\mathrm{cond}}^{\mathrm{Drift}}$ and $\mathcal{L}_{\mathrm{cond}}^{\mathrm{Leap}}$ are defined in Eqs. equation 5 and equation 7, respectively.

Inference with Classifier-Free Guidance. At inference, conditional and unconditional predictions are evaluated at the same state and time:

$$g_m(z_t, t \mid I) = (1 - \omega)\mathcal{F}_{\theta_m}(z_t, t \mid \varnothing) + \omega\mathcal{F}_{\theta_m}(z_t, t \mid I), \tag{12}$$

where ω is the guidance scale. Since the network outputs have different meanings for the two parameterizations, the corresponding guided velocity is

$$\begin{aligned} u_m(z_t, t \mid I) &= (1 - \omega)v_m(z_t, t \mid \varnothing) + \omega v_m(z_t, t \mid I) \\ &= \begin{cases} g_{\mathrm{Drift}}(z_t, t \mid I), & m = \mathrm{Drift}, \\ \dfrac{g_{\mathrm{Leap}}(z_t, t \mid I) - z_t}{r(t)}, & m = \mathrm{Leap}. \end{cases} \end{aligned} \tag{13}$$

For FakeFlow$_{\mathrm{Drift}}$, g_{Drift} is directly a velocity; for FakeFlow$_{\mathrm{Leap}}$, g_{Leap} is converted from a target-label prediction to a velocity.

Because both branches share the same state and denominator, applying CFG before or after the target-to-velocity conversion is equivalent.

For a test image I, we sample an initial state $z_0 \sim \mathcal{N}(0, \mathbf{I}_C)$ and set $z_m^{(0)} = z_0$. We then discretize $0 = t_0 < \cdots < t_S = 1$ and perform Euler updates:

$$z_m^{(k+1)} = z_m^{(k)} + \Delta t_k \, u_m\big(z_m^{(k)}, t_k \mid I\big), \qquad \Delta t_k = t_{k+1} - t_k. \tag{14}$$

By default, we use one-step inference with $S = 1$, $t_0 = 0$, and $\Delta t_0 = 1$. Since $r(0) = 1$,

$$\begin{aligned} z_{\text{Drift}}^{(1)} &= z_0 + g_{\text{Drift}}(z_0, 0 \mid I), \\ z_{\text{Leap}}^{(1)} &= g_{\text{Leap}}(z_0, 0 \mid I). \end{aligned} \tag{15}$$

Thus, FakeFlow$_{\text{Drift}}$ updates the initial state with the guided velocity, whereas FakeFlow$_{\text{Leap}}$ directly yields the guided target prediction. The final class is

$$y_{\text{pred}}^m = \underset{j \in \{1,\ldots,C\}}{\arg\max} \big[z_m^{(S)}\big]_j, \qquad m \in \{\text{Drift}, \text{Leap}\}. \tag{16}$$

4.4 Theoretical Justification

Following conditional Flow Matching (Lipman et al., 2023), we formalize authenticity classification in label space and analyze the effect of CFG. Let $\eta(I) = \mathbb{E}[\mathbf{e}_y \mid I]$. Denote the one-step estimate by $\hat{z}_m(z_0 \mid c)$: $z_0 + F_{\text{Drift}}(z_0, 0 \mid c)$ for Drift and $F_{\text{Leap}}(z_0, 0 \mid c)$ for Leap.

Definition 4.1 (Label Flow Risk)**.** For $m \in \{\text{Drift}, \text{Leap}\}$, define

$$\mathcal{R}^m(F) = \frac{1}{2}\mathbb{E}\left[w_m(t)\|F(z_t, t \mid c) - Y_m\|_2^2\right], \tag{17}$$

where $(Y_{\text{Drift}}, w_{\text{Drift}}) = (\mathbf{e}_y - z_0, 1)$ and $(Y_{\text{Leap}}, w_{\text{Leap}}) = (\mathbf{e}_y, r(t)^{-2})$. The condition c is independently dropped with probability p_{drop}.

Remark 4.2. At $t = 0$ and $p_{\text{drop}} = 0$, both risks reduce to $\frac{1}{2}\mathbb{E}\|\hat{z}_m - \mathbf{e}_y\|_2^2$: one-hot squared regression, equivalent to the Brier loss for probability-valued outputs (Brier, 1950; Gneiting & Raftery, 2007).

Remark 4.2 connects FakeFlow to label regression, while its time-dependent objective extends supervision along the label path.

Lemma 4.3 (Label Flow Transport)**.** *Under the sampling and regularity assumptions in Appendix A.1, the population minimizers satisfy*

$$F_{\text{Drift}}^*(z, t \mid I) = \mathbb{E}[\mathbf{e}_y - z_0 \mid z_t = z, I].$$

When $1 - t \geq \epsilon$, $(F_{\text{Leap}}^* - z)/r(t) = F_{\text{Drift}}^*$. *The optimal Drift field transports the initial Gaussian to* $\sum_j p(y = j \mid I)\delta_{\mathbf{e}_j}$ *in the weak limit* $t \uparrow 1$.

Lemma 4.3 (proof in Appendix A.1) formalizes classification as image-conditioned label transport, with velocity and endpoint prediction providing two corresponding parameterizations.

Corollary 4.4 (Bayes-Optimal One-Step Decoding)**.** *Under Lemma 4.3 and the zero-time continuity conditions in Appendix A.2, setting* $\omega = 1$ *yields* $z_m^{(1)} = \eta(I)$ *for both variants. Thus, maximum-coordinate decoding gives* $\arg\max_j p(y = j \mid I)$.

Corollary 4.4 (proof in Appendix A.2) establishes compatibility between population-optimal label-flow prediction and Bayes classification (Devroye et al., 1996).

Definition 4.5 (Initial-State Fluctuation)**.** Let $\mathbf{a} = \mathbf{e}_2 - \mathbf{e}_1$, $s_{\text{c}} = \mathbf{a}^\top \hat{z}_m(z_0 \mid I)$, and $s_\varnothing = \mathbf{a}^\top \hat{z}_m(z_0 \mid \varnothing)$. Over z_0, define $\mu_{\text{c}}(I) = \mathbb{E}[s_{\text{c}}]$, $\sigma_{\text{c}}^2 = \text{Var}(s_{\text{c}})$, $\sigma_\varnothing^2 = \text{Var}(s_\varnothing)$, $\kappa = \text{Cov}(s_{\text{c}}, s_\varnothing)$, and $\rho = \kappa/(\sigma_{\text{c}}\sigma_\varnothing)$.

Lemma 4.6 (Guidance as a Control Variate)**.** *Assume finite second moments and image-independent* σ_{c}^2 *and* κ. *For* $\omega > 0$ *and* $\lambda = 1 - \omega^{-1}$, *the normalized guided score preserves inter-image mean differences and has variance*

$$V(\omega) = \sigma_{\text{c}}^2 - 2\lambda\kappa + \lambda^2\sigma_\varnothing^2. \tag{18}$$

If $\kappa > 0$, *some* $\omega > 1$ *gives* $V(\omega) < V(1)$. *If* $0 < \kappa < \sigma_\varnothing^2$, *its minimum occurs at* $\omega^* = \sigma_\varnothing^2/(\sigma_\varnothing^2 - \kappa)$, *with* $V(\omega^*) = (1 - \rho^2)\sigma_{\text{c}}^2$.

Lemma 4.6 (proof in Appendix A.3) identifies the unconditional branch as a control variate (Glasserman, 2004): appropriate guidance reduces initial-state fluctuations while preserving mean score separation.

Corollary 4.7 (Guided Mis-ranking Bound)**.** *Under Lemma 4.6, consider generated and real images with independent initial states and* $\Delta = \mu_c(I_f) - \mu_c(I_r) > 0$*. Then*

$$P(s_\omega(I_f) \le s_\omega(I_r)) \le \frac{2V(\omega)}{2V(\omega) + \Delta^2}. \tag{19}$$

Corollary 4.7 (proof in Appendix A.4) shows that variance reduction tightens the mis-ranking bound, providing a theoretical explanation for CFG's potential benefit to authenticity discrimination.

5 Experiment

5.1 Experimental Settings

We train all FakeFlow models exclusively on the Stable Diffusion v1.4 (SD-v1.4) training split of GenImage (Zhu et al., 2023) and evaluate them on four benchmarks: eight generator subsets from GenImage, ten from AIGI-Holmes (Zhou et al., 2025b), 18 pixel-level (Pix.) and semantic (Sem.) subsets covering nine generators from AIGI-Now (Chen et al., 2025a), and the In-the-Wild benchmark comprising Chameleon (Yan et al., 2024), WildRF (Cavia et al., 2024), SocialRF, and CommunityAI (Li et al., 2025b). We report the mean binary classification accuracy across test subsets for each benchmark. We compare FakeFlow with SAFE (Li et al., 2025a), Effort (Yan et al., 2025), DDA (Chen et al., 2025b), OMAT (Zhou et al., 2025a), AIDE (Yan et al., 2024), OmniAID (Guo et al., 2026), SimLBR (Dhakal et al., 2026), CLIP-ADA (Deng et al., 2026), and SFAH (Wang et al., 2026). We additionally construct a DINOv3-Linear baseline by training a linear classifier on frozen DINOv3 features (Siméoni et al., 2025), and compare it with FakeFlow$_{\mathrm{Drift}}$ and FakeFlow$_{\mathrm{Leap}}$.

Implementation details. We use a frozen DINOv3 ViT-7B/16 (Siméoni et al., 2025) as the visual encoder. During training, all backbone parameters remain frozen, and only the visual token projection layer, label-state encoder, time encoder, cross-attention module, and output head are updated. Input images are directly resized to 224×224 using bicubic interpolation, converted to floating-point tensors, and normalized using the ImageNet mean $(0.485, 0.456, 0.406)$ and standard deviation $(0.229, 0.224, 0.225)$. Both FakeFlow$_{\mathrm{Drift}}$ and FakeFlow$_{\mathrm{Leap}}$ are trained for 5 epochs using AdamW with a learning rate of 1×10^{-3}. FakeFlow$_{\mathrm{Drift}}$ uses a per-process batch size of 128 with automatic mixed precision enabled, whereas FakeFlow$_{\mathrm{Leap}}$ uses a per-process batch size of 64 without automatic mixed precision. Both models are trained with mean squared error loss.

5.2 Cross-Domain Generalization

Overall Performance. FakeFlow achieves competitive performance against recent AI-generated image detectors across four benchmark suites. On GenImage (Table 1), FakeFlow$_{\mathrm{Leap}}$ and FakeFlow$_{\mathrm{Drift}}$ achieve macro accuracies of 0.975 and 0.971, exceeding the strongest competing detector, OMAT (0.946), by 2.9 and 2.5 percentage points, respectively. Leap achieves the highest accuracy on BigGAN, while Drift leads on VQDM and Wukong. On In-the-Wild (Table 2), Drift and Leap achieve macro accuracies of 0.949 and 0.928, surpassing DDA (0.850), the strongest competing detector, by 9.9 and 7.8 percentage points. Both variants outperform the competing dedicated detectors on all four real-world datasets. On AIGI-Holmes (Table 3), Leap and Drift reach macro accuracies of 0.982 and 0.978, improving over AIDE (0.970) by 1.2 and 0.8 percentage points. Drift achieves the highest accuracy on Janus, JP-1B, and JP-7B, and both variants attain 0.978 on VAR. On AIGI-Now (Table 4), Leap and Drift achieve macro accuracies of 0.916 and 0.887, exceeding the strongest competing detector, DDA (0.695), by 22.1 and 19.2 percentage points. Notably, Leap ranks first on all nine semantically discriminative subsets, highlighting its strong performance on semantic discrimination. Overall, Leap achieves higher macro accuracy on the three generator-based benchmarks, while Drift performs better on the real-world benchmark suite.

Table 1: Generalization on GenImage. Accuracy on eight generator subsets. The best results are highlighted in bold.

Method	ADM	BigGAN	Midjourney	VQDM	GLIDE	SD-v1.4	SD-v1.5	Wukong	Avg.
SAFE$_{KDD2025}$	0.600	0.880	0.829	0.708	0.836	0.964	0.967	0.903	0.836
Effort$_{ICML2025}$	0.704	0.749	0.756	0.824	0.807	0.874	0.873	0.867	0.807
DDA$_{NeurIPS2025}$	0.880	0.741	**0.960**	0.719	0.862	0.986	0.985	0.986	0.890
OMAT$_{NeurIPS2025}$	0.837	0.973	0.903	0.954	0.974	0.974	0.973	0.975	0.946
AIDE$_{ICLR2025}$	0.602	0.648	0.813	0.694	0.514	0.959	0.957	0.954	0.768
OmniAID$_{ICML2026}$	0.809	0.992	0.826	0.920	**0.994**	0.997	**0.996**	0.989	0.940
SimLBR$_{CVPR2026}$	**0.970**	0.880	0.917	0.935	0.924	0.983	0.981	0.974	0.945
CLIP-ADA$_{TIFS2026}$	0.674	0.867	0.890	0.773	0.742	0.950	0.947	0.880	0.840
SFAH$_{TIFS2026}$	0.855	0.953	0.959	0.912	0.909	0.993	0.991	0.983	0.944
FakeFlow									
Baseline	0.849	0.991	0.934	0.992	0.963	**0.998**	**0.996**	0.994	0.964
FakeFlow$_{\text{Drift}}$	0.895	0.992	0.912	**0.996**	0.983	0.996	0.995	**0.995**	0.971
FakeFlow$_{\text{Leap}}$	0.908	**0.994**	0.943	0.993	0.985	0.995	0.993	0.993	**0.975**

Table 2: Generalization on In-the-Wild. Accuracy on four real-world datasets and their macro average.

Method	Chameleon	WildRF	SocialRF	CommunityAI	Avg.
SAFE$_{KDD2025}$	0.520	0.613	0.548	0.532	0.556
Effort$_{ICML2025}$	0.588	0.567	0.523	0.533	0.553
DDA$_{NeurIPS2025}$	0.824	0.904	0.832	0.847	0.850
OMAT$_{NeurIPS2025}$	0.629	0.674	0.607	0.634	0.636
AIDE$_{ICLR2025}$	0.574	0.584	0.560	0.542	0.565
OmniAID$_{ICML2026}$	0.810	0.673	0.573	0.828	0.721
FakeFlow					
Baseline	0.914	**0.961**	**0.943**	0.948	0.940
FakeFlow$_{\text{Drift}}$	**0.953**	0.952	0.919	**0.970**	**0.949**
FakeFlow$_{\text{Leap}}$	0.910	0.940	0.916	0.946	0.928

Table 3: Generalization on AIGI-Holmes. Accuracy on ten recent generators. The best results are highlighted in bold.

Method	FLUX	Infinity	Janus	JP-1B	JP-7B	LlamaGen	PixArt-XL	SD3.5-L	Show-o	VAR	Avg.
SAFE$_{KDD2025}$	0.932	0.965	0.482	0.482	0.483	0.913	0.896	0.940	0.962	0.911	0.796
Effort$_{ICML2025}$	0.794	0.804	0.483	0.650	0.569	0.798	0.804	0.765	0.793	0.764	0.722
DDA$_{NeurIPS2025}$	**0.972**	0.989	0.985	0.993	0.987	0.993	0.994	0.970	0.948	0.804	0.963
OMAT$_{NeurIPS2025}$	0.947	0.955	0.651	0.756	0.641	0.967	0.969	0.957	0.969	0.959	0.877
AIDE$_{ICLR2025}$	0.944	0.987	0.912	0.989	0.978	0.994	0.986	**0.994**	0.980	0.936	0.970
OmniAID$_{ICML2026}$	0.881	0.987	0.558	0.627	0.544	0.975	0.969	0.743	0.870	0.938	0.809
FakeFlow											
Baseline	0.933	**0.998**	0.996	0.995	0.986	**0.999**	**0.999**	0.891	**0.997**	0.922	0.972
FakeFlow$_{\text{Drift}}$	0.947	**0.998**	**0.998**	**0.996**	**0.994**	0.997	0.998	0.879	0.992	**0.978**	0.978
FakeFlow$_{\text{Leap}}$	0.955	0.994	0.994	0.992	0.993	0.993	0.994	0.941	0.992	**0.978**	**0.982**

Table 4: Generalization on AIGI-Now. Accuracy on pixel-artifact-discriminative (Pix.) and semantically discriminative (Sem.) subsets from nine generators.

Method	FLUX-dev		FLUX-kera		FLUX-kontext		FLUX-pro		GPT-4o		Jimeng		Keling		MiniMax		Nano		Avg.
	Pix.	*Sem.*	*Pix.*	*Sem.*	*Pix.*	*Sem.*	*Pix.*	*Sem.*	*Pix.*	*Sem.*	*Pix.*	*Sem.*	*Pix.*	*Sem.*	*Pix.*	*Sem.*	*Pix.*	*Sem.*	
SAFE$_{KDD2025}$	0.903	0.490	0.532	0.492	0.831	0.494	0.521	0.486	0.977	0.488	0.509	0.486	0.960	0.487	0.590	0.491	0.951	0.484	0.621
Effort$_{ICML2025}$	0.789	0.679	0.796	0.669	0.728	0.610	0.688	0.690	0.753	0.580	0.522	0.555	0.782	0.677	0.772	0.687	0.796	0.657	0.690
DDA$_{NeurIPS2025}$	0.916	0.512	0.594	0.499	0.827	0.529	0.766	0.550	0.923	0.654	0.870	0.654	0.961	0.646	0.833	0.505	0.816	0.562	0.695
OMAT$_{NeurIPS2025}$	0.911	0.475	0.649	0.469	0.847	0.507	0.591	0.515	0.744	0.452	0.491	0.465	0.936	0.526	0.699	0.467	0.891	0.468	0.615
AIDE$_{ICLR2025}$	**0.991**	0.590	0.504	0.569	**0.979**	0.806	0.601	0.538	0.747	0.518	0.639	0.514	**0.982**	0.554	0.514	0.541	**0.989**	0.518	0.672
OmniAID$_{ICML2026}$	0.825	0.506	0.534	0.496	0.759	0.506	0.666	0.504	0.577	0.490	0.580	0.505	0.758	0.499	0.647	0.495	0.772	0.505	0.590
FakeFlow																			
Baseline	0.944	0.962	0.846	0.811	0.730	0.756	0.813	0.948	0.898	0.960	0.824	0.940	0.884	0.913	0.727	0.784	0.898	0.922	0.864
FakeFlow$_{\text{Drift}}$	0.947	0.969	0.900	0.820	0.760	0.765	0.848	0.948	0.958	0.971	0.854	0.934	0.935	0.914	0.781	0.779	0.972	0.908	0.887
FakeFlow$_{\text{Leap}}$	0.963	**0.984**	**0.933**	**0.908**	0.778	**0.820**	**0.865**	**0.971**	0.959	**0.987**	**0.879**	**0.969**	0.949	**0.943**	0.803	**0.846**	0.973	**0.963**	**0.916**

5.3 Ablation Study

Effect of Loss Functions. Table 5 compares L1, Smooth L1, and MSE losses. MSE achieves the highest accuracy for both variants on GenImage, AIGI-Holmes, and AIGI-Now. On In-the-Wild,

Table 5: Loss ablation results.

Method	Loss	Gen.	Holmes	Now	Wild
FakeFlow$_{\text{Leap}}$	L1	0.9687	0.9759	0.8973	0.9416
FakeFlow$_{\text{Leap}}$	Smooth L1	0.9726	0.9798	0.9084	0.9448
FakeFlow$_{\text{Leap}}$	**MSE**	0.9754	0.9824	0.9160	0.9278
FakeFlow$_{\text{Drift}}$	L1	0.9663	0.9738	0.8719	0.9425
FakeFlow$_{\text{Drift}}$	Smooth L1	0.9689	0.9764	0.8827	0.9463
FakeFlow$_{\text{Drift}}$	**MSE**	0.9705	0.9777	0.8865	0.9487

Table 6: Module ablation results.

Method	Gen.	Holmes	Now	Wild
Baseline	0.9640	0.9720	0.8640	0.9400
FakeFlow$_{\text{Drift}}$ w/o CFG	0.9672	0.9745	0.8762	0.9443
FakeFlow$_{\text{Leap}}$ w/o CFG	0.9698	0.9773	0.8912	0.9336
FakeFlow$_{\text{Drift}}$	0.9705	0.9777	0.8865	**0.9487**
FakeFlow$_{\text{Leap}}$	**0.9754**	**0.9824**	**0.9160**	0.9278

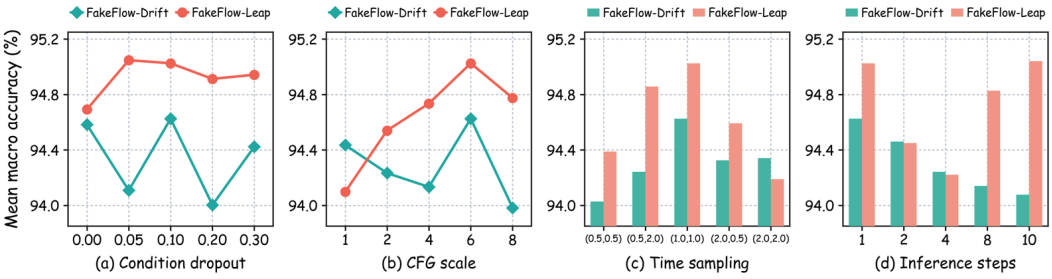

Figure 3: Hyperparameter sensitivity of FakeFlow$_{\text{Drift}}$ and FakeFlow$_{\text{Leap}}$.

MSE performs best for Drift, whereas Smooth L1 yields the highest accuracy for Leap. These results support MSE as the default loss while showing that its advantage depends on the parameterization and benchmark.

Effect of Model Components. As shown in Table 6, label-space trajectory modeling improves performance across most benchmarks. CFG further improves both variants on GenImage, AIGI-Holmes, and AIGI-Now, while its effect on In-the-Wild depends on the parameterization. Specifically, Drift benefits from CFG on In-the-Wild, whereas Leap exhibits a slight decrease in accuracy. This contrast indicates that the gains from CFG are not uniform across benchmarks and parameterizations. Overall, these results support the effectiveness of label-space trajectory modeling while highlighting the varying contribution of CFG.

5.4 Sensitivity Analysis

Figure 3 evaluates the sensitivity of FakeFlow$_{\text{Drift}}$ and FakeFlow$_{\text{Leap}}$ to key hyperparameters, using the average accuracy across the four benchmarks. For each variant, performance varies by less than one percentage point within the tested ranges, suggesting relatively low hyperparameter sensitivity. A moderate condition-dropout rate provides a favorable balance, with $p_{\text{drop}} = 0.1$ performing well for both variants. Both models achieve their highest accuracy among the tested CFG scales at $\omega = 6$, whereas increasing the scale to 8 reduces performance. Uniform time sampling, corresponding to $(\alpha, \beta) = (1.0, 1.0)$, also gives the best results among the evaluated distributions. The effect of additional inference steps is not consistently positive: Drift performs best with one step, while Leap shows non-monotonic changes and only a marginal difference between one and ten steps. Based on these results, we use $(1.0, 1.0)$ for time sampling, $p_{\text{drop}} = 0.1$, a guidance scale of 6, and one-step inference by default. This configuration balances detection performance and computational cost.

6 Conclusion

This paper presents FakeFlow, a generative classification framework that formulates AI-generated image detection as an image-conditioned transport process in label space. We introduce two complementary parameterizations, FakeFlow$_{\text{Drift}}$ and FakeFlow$_{\text{Leap}}$, which model trajectory velocities and target labels, respectively, and incorporate CFG to strengthen image-specific evidence during prediction. Extensive experiments demonstrate strong generalization across unseen generators and real-world scenarios, suggesting that modeling the decision process itself provides a promising direction for generalizable AI-generated image detection.

AI USE STATEMENT.

Generative AI tools were used exclusively to assist with language editing, including improvements to grammar, clarity, and readability. They were not used to generate research ideas, design methods or experiments, analyze data, interpret results, or formulate scientific claims. The authors reviewed all AI-assisted revisions and take full responsibility for the final manuscript and its scientific content.

ETHICS STATEMENT

This work studies AI-generated image detection using existing image benchmarks and pretrained visual representations. It does not involve human-subject experiments or the collection of new personal data. We do not identify additional ethical concerns specific to the research procedures in this study.

REPRODUCIBILITY STATEMENT

Our code is available at `https://anonymous.4open.science/r/fakeflow-7575`.

REFERENCES

Glenn W. Brier. Verification of forecasts expressed in terms of probability. *Monthly Weather Review*, 78(1):1–3, 1950. doi: 10.1175/1520-0493(1950)078⟨0001:VOFEIT⟩2.0.CO;2.

Bar Cavia, Eliahu Horwitz, Tal Reiss, and Yedid Hoshen. Real-time deepfake detection in the real-world. *arXiv preprint arXiv:2406.09398*, 2024. doi: 10.48550/arXiv.2406.09398. URL https://arxiv.org/abs/2406.09398.

Ruoxin Chen, Jiahui Gao, Kaiqing Lin, Keyue Zhang, Yandan Zhao, Isabel Guan, Taiping Yao, and Shouhong Ding. Task-model alignment: A simple path to generalizable AI-generated image detection. *arXiv preprint arXiv:2512.06746*, 2025a.

Ruoxin Chen, Junwei Xi, Zhiyuan Yan, et al. Dual data alignment makes AI-generated image detector easier generalizable. *arXiv preprint arXiv:2505.14359*, 2025b.

Jingyi Deng, Chong Zhang, Chenken Xu, Chenhao Lin, Zhengyu Zhao, Shuai Liu, Qian Wang, and Chao Shen. Clip-ada: Clip-guided artifact-invariant generalizable synthetic image detection. *IEEE Transactions on Information Forensics and Security*, 21:3588–3601, 2026. doi: 10.1109/TIFS.2026.3676657.

Luc Devroye, László Györfi, and Gábor Lugosi. *A Probabilistic Theory of Pattern Recognition*, volume 31 of *Stochastic Modelling and Applied Probability*. Springer, New York, 1996. doi: 10.1007/978-1-4612-0711-5.

Aayush Dhakal, Subash Khanal, Srikumar Sastry, Jacob Arndt, Philipe Ambrozio Dias, Dalton Lunga, and Nathan Jacobs. SimLBR: Learning to detect fake images by learning to detect real images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2026.

Paul Glasserman. *Monte Carlo Methods in Financial Engineering*, volume 53 of *Stochastic Modelling and Applied Probability*. Springer, New York, 2004. doi: 10.1007/978-0-387-21617-1.

Tilmann Gneiting and Adrian E. Raftery. Strictly proper scoring rules, prediction, and estimation. *Journal of the American Statistical Association*, 102(477):359–378, 2007. doi: 10.1198/016214506000001437.

Yuncheng Guo, Junyan Ye, Chenjue Zhang, Hengrui Kang, Haohuan Fu, Conghui He, and Weijia Li. OmniAID: Decoupling semantic and artifacts for universal AI-generated image detection in the wild. In *Proceedings of the International Conference on Machine Learning*, 2026. URL https://arxiv.org/abs/2511.08423.

Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022. URL https://arxiv.org/abs/2207.12598.

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

Tianhong Li and Kaiming He. Back to basics: Let denoising generative models denoise. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 36115–36125, June 2026.

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

Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matthew Le. Flow matching for generative modeling. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=PqvMRDCJT9t.

Zhengzhe Liu, Xiaojuan Qi, and Philip H. S. Torr. Global texture enhancement for fake face detection in the wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8060–8069, 2020.

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.

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 *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 8748–8763. PMLR, 2021.

Oriane Siméoni, Huy V. Vo, Maximilian Seitzer, et al. Dinov3. *arXiv preprint arXiv:2508.10104*, 2025.

Chuangchuang Tan, Yao Zhao, Shikui Wei, Guanghua Gu, Ping Liu, and Yunchao Wei. Frequency-aware deepfake detection: Improving generalizability through frequency space domain learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 5052–5060, 2024a.

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

Jun Wang, Jian Zou, Zitong Yu, Shuo Ye, Wenbo Wan, Lingchen Gu, Jiantao Zhou, and Weiming Zhang. Suspicious frequency amplified hybrid framework for generalizable AIGC image detection. *IEEE Transactions on Information Forensics and Security*, 21:7407–7422, 2026. doi: 10.1109/TIFS.2026.3723118.

Sheng-Yu Wang, Oliver Wang, Richard Zhang, Andrew Owens, and Alexei A. Efros. CNN-generated images are surprisingly easy to spot... for now. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8695–8704, 2020.

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

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 *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 70268–70288. PMLR, 2025.

Zheng Yang, Ruoxin Chen, Zhiyuan Yan, Ke-Yue Zhang, Xinghe Fu, Shuang Wu, Xiujun Shu, Taiping Yao, Shouhong Ding, Zequn Qin, and Xi Li. All patches matter, more patches better: Enhance AI-generated image detection via panoptic patch learning. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=ob7PJs8kPU.

Yue Zhou, Xinan He, Kaiqing Lin, Bing Fan, Feng Ding, and Bin Li. Breaking latent prior bias in detectors for generalizable aigc image detection. *arXiv preprint arXiv:2506.00874*, 2025a.

Yue Zhou, Xinan He, Kaiqing Lin, Bing Fan, Feng Ding, and Bin Li. Simplicity prevails: The emergence of generalizable aigi detection in visual foundation models. *arXiv preprint arXiv:2602.01738*, 2026.

Ziyin Zhou, Yunpeng Luo, Yuanchen Wu, Ke Sun, Jiayi Ji, Ke Yan, Shouhong Ding, Xiaoshuai Sun, Yunsheng Wu, and Rongrong Ji. AIGI-Holmes: Towards explainable and generalizable AI-generated image detection via multimodal large language models. *arXiv preprint arXiv:2507.02664*, 2025b.

Mingjian Zhu, Hanting Chen, Qiangyu Yan, Xudong Huang, Guanyu Lin, Wei Li, Zhiyong Tu, Hailin Hu, Jie Hu, and Yunhe Wang. GenImage: A million-scale benchmark for detecting AI-generated image. In *Advances in Neural Information Processing Systems*, volume 36, pp. 77771–77782, 2023.

A THEORETICAL JUSTIFICATION

This appendix follows the notation in the main text and proves Lemma 4.3, Corollary 4.4, Lemma 4.6, and Corollary 4.7, in order. The initial state $z_0 \sim \mathcal{N}(0, \mathbf{I}_C)$, time sampling, and condition dropout are mutually independent and independent of (I, y); the null condition is denoted by $\varnothing$. For ODE integration and one-step inference, we use the explicitly specified smooth conditional-expectation versions of the population-optimal predictors below, together with their continuous extensions at $t = 0$.

A.1 PROOF OF LEMMA 4.3

Proof. Following the definition of Label Flow Risk in the main text, let $U = (z_t, t, c)$ and $M_m(U) = \mathbb{E}[Y_m \mid U]$, where $Y_{\text{Drift}} = \mathbf{e}_y - z_0$ and $Y_{\text{Leap}} = \mathbf{e}_y$. For any predictor F with finite risk,

$$\|F(U) - Y_m\|_2^2 = \|F(U) - M_m(U)\|_2^2 + \|M_m(U) - Y_m\|_2^2 + 2\langle F(U) - M_m(U), M_m(U) - Y_m\rangle.$$

Since $w_m(t)$ and $F(U) - M_m(U)$ are measurable functions of U, and $\mathbb{E}[M_m(U) - Y_m \mid U] = 0$, the weighted expectation of the cross term vanishes. Therefore, the risk in the main text admits the decomposition

$$\mathcal{R}^m(F) - \mathcal{R}^m(M_m) = \frac{1}{2}\mathbb{E}\big[w_m(t)\|F(U) - M_m(U)\|_2^2\big] \geq 0. \tag{20}$$

Here, $w_{\text{Drift}}(t) = 1$ and $w_{\text{Leap}}(t) = r(t)^{-2} > 0$. Hence, equality holds if and only if $F(U) = M_m(U)$ almost surely: positive time-dependent weights do not change the pointwise risk minimizer. Moreover, since $\pi(t)$ has positive density on $(0, 1)$ and $0 < p_{\text{drop}} < 1$, both conditional and unconditional branches are identified by the training risk. Thus, for almost every $t \in (0, 1)$,

$$\begin{aligned} F^*_{\text{Drift}}(z, t \mid c) &= \mathbb{E}[\mathbf{e}_y - z_0 \mid z_t = z, t, c], \\ F^*_{\text{Leap}}(z, t \mid c) &= \mathbb{E}[\mathbf{e}_y \mid z_t = z, t, c]. \end{aligned} \tag{21}$$

To express these minimizers explicitly, fix a condition c and let $q_j(c) = P(y = j \mid c)$. By the linear path in Eq. (1) and the independence of the initial state,

$$p_t(z \mid y = j, c) = \varphi_t(z - t\mathbf{e}_j), \qquad \varphi_t(u) = \frac{\exp[-\|u\|_2^2/(2(1-t)^2)]}{(2\pi)^{C/2}(1-t)^C}.$$

Here, φ_t is the density of $\mathcal{N}(0, (1-t)^2\mathbf{I}_C)$. The law of total probability and Bayes' rule give

$$\begin{aligned} p_t(z \mid c) &= \sum_{j=1}^{C} q_j(c)\varphi_t(z - t\mathbf{e}_j) > 0, \\ \beta_j(z, t \mid c) &:= P(y = j \mid z_t = z, t, c) = \frac{q_j(c)\varphi_t(z - t\mathbf{e}_j)}{\sum_{k=1}^{C} q_k(c)\varphi_t(z - t\mathbf{e}_k)}, \\ F^*_{\text{Leap}}(z, t \mid c) &= \sum_{j=1}^{C} \beta_j(z, t \mid c)\mathbf{e}_j. \end{aligned} \tag{22}$$

Since $\|\mathbf{e}_j\|_2 = 1$, canceling the common factors in the numerator and denominator also yields

$$\beta_j(z, t \mid c) = \frac{q_j(c)\exp\{tz_j/(1-t)^2\}}{\sum_{k=1}^{C} q_k(c)\exp\{tz_k/(1-t)^2\}}.$$

The denominator is strictly positive for all $z \in \mathbb{R}^C$ and $t < 1$. Consequently, these expressions are smooth in $(z, t) \in \mathbb{R}^C \times [0, 1)$.

By Eq. (1), the following identity holds for each sample when $t < 1$:

$$\mathbf{e}_y - z_0 = \frac{\mathbf{e}_y - z_t}{1 - t}.$$

Substituting this identity into Eq. (21) gives

$$F^*_{\text{Drift}}(z, t \mid c) = \frac{F^*_{\text{Leap}}(z, t \mid c) - z}{1 - t}. \tag{23}$$

In particular, when $1 - t \geq \epsilon$, we have $r(t) = 1 - t$, and therefore

$$\frac{F^*_{\text{Leap}}(z, t \mid I) - z}{r(t)} = F^*_{\text{Drift}}(z, t \mid I),$$

establishing the relationship between the two parameterizations stated in the lemma.

We next show that this velocity field preserves the constructed conditional marginal distributions. Fix I, write $B(z,t) = F^*_{\text{Leap}}(z, t \mid I)$ and $v^*(z,t) = F^*_{\text{Drift}}(z, t \mid I)$, and regard $\beta(z, t \mid I)$ as a column vector. Differentiating the explicit posterior gives

$$\nabla_z B(z,t) = \frac{t}{(1-t)^2} \left(\text{diag}(\beta) - \beta\beta^\top\right).$$

For any $h \in \mathbb{R}^C$,

$$0 \leq h^\top \left(\text{diag}(\beta) - \beta\beta^\top\right) h = \sum_j \beta_j h_j^2 - \left(\sum_j \beta_j h_j\right)^2 \leq \|h\|_2^2.$$

Thus, the operator norm of the matrix in parentheses is at most 1. Furthermore, $B(z,t)$ is a probability vector, so $\|B(z,t)\|_2 \leq 1$. By Eq. (23), for any $0 < \delta < 1$ and $t \in [0, 1-\delta]$,

$$\|\nabla_z v^*(z,t)\|_{\text{op}} \leq \delta^{-1} + \delta^{-3}, \qquad \|v^*(z,t)\|_2 \leq \delta^{-1}(1 + \|z\|_2). \tag{24}$$

Hence, on each such time interval, v^* is globally Lipschitz in the state and has at most linear growth. The ODE $dZ_t/dt = v^*(Z_t, t)$ with $Z_0 = z_0$ therefore has a unique solution on $[0, 1-\delta]$. Uniqueness ensures that solutions on overlapping intervals coincide, yielding a unique solution on $[0, 1)$.

For a fixed label j, define $X_t^j = (1-t)z_0 + t\mathbf{e}_j$, with conditional velocity $u_j(z,t) = (\mathbf{e}_j - z)/(1-t)$. For any smooth, compactly supported test function ϕ, the chain rule gives

$$\begin{aligned}\frac{d}{dt}\mathbb{E}[\phi(X_t^j)] &= \mathbb{E}[\nabla\phi(X_t^j)^\top(\mathbf{e}_j - z_0)] \\ &= \int \nabla\phi(z)^\top u_j(z,t)\, p_t(z \mid y = j, I)\, dz.\end{aligned}$$

Thus, $p_t(z \mid y = j, I)$ and u_j satisfy the conditional continuity equation. Taking the weighted sum with $q_j(I) = P(y = j \mid I)$ and using Eq. (22), we obtain

$$\begin{aligned}\sum_j q_j(I) p_t(z \mid y = j, I) u_j(z,t) &= p_t(z \mid I) \sum_j \beta_j(z, t \mid I) \frac{\mathbf{e}_j - z}{1-t} \\ &= p_t(z \mid I) v^*(z,t).\end{aligned}$$

Consequently, the marginal density satisfies

$$\partial_t p_t(z \mid I) + \nabla_z \cdot \left(p_t(z \mid I) v^*(z,t)\right) = 0. \tag{25}$$

The distribution obtained by the ODE pushforward also satisfies Eq. (25), and both distributions equal $\mathcal{N}(0, \mathbf{I}_C)$ at $t = 0$. To establish uniqueness for the same initial distribution, fix $\tau < 1$ and let $\Phi_{s,\tau}$ denote the flow map from s to τ. For any terminal test function ϕ, set $\psi_s(z) = \phi(\Phi_{s,\tau}(z))$. Then

$$\partial_s \psi_s + v^* \cdot \nabla \psi_s = 0, \qquad \int \phi\, d\nu_\tau = \int \psi_0\, d\nu_0$$

holds for any weak solution ν of the continuity equation. The regularity in Eq. (24) justifies this characteristic argument, so the initial distribution uniquely determines ν_τ. Therefore, for every $t \in [0, 1)$,

$$\text{Law}(Z_t \mid I) = p_t(\cdot \mid I) = \text{Law}(z_t \mid I),$$

where $p_t(\cdot \mid I)$ in the middle denotes the corresponding probability distribution.

Finally, since $z_t = t\mathbf{e}_y + (1-t)z_0$, we have $z_t \to \mathbf{e}_y$ almost surely as $t \to 1$. For any bounded continuous function ϕ, dominated convergence and the equality of marginal distributions imply

$$\lim_{t \uparrow 1} \mathbb{E}[\phi(Z_t) \mid I] = \lim_{t \uparrow 1} \mathbb{E}[\phi(z_t) \mid I] = \sum_{j=1}^C P(y = j \mid I)\phi(\mathbf{e}_j).$$

This is precisely the weak convergence of $\text{Law}(Z_t \mid I)$ to $p(\mathbf{e}_y \mid I) = \sum_j P(y = j \mid I)\delta_{\mathbf{e}_j}$, completing the proof. □

A.2 Proof of Corollary 4.4

Proof. By Lemma 4.3 and Eq. (22), the density factor $\varphi_0(z)$ at $t = 0$ is independent of the class j. Hence,

$$\beta_j(z, 0 \mid c) = \frac{q_j(c)\varphi_0(z)}{\sum_k q_k(c)\varphi_0(z)} = q_j(c).$$

Combining this with Eq. (23) yields

$$F^*_{\text{Leap}}(z, 0 \mid c) = \mathbb{E}[\mathbf{e}_y \mid c], \qquad F^*_{\text{Drift}}(z, 0 \mid c) = \mathbb{E}[\mathbf{e}_y \mid c] - z.$$

By the definitions of $\eta(I)$ and $\bar{\eta}$ and the independence of condition dropout, $\mathbb{E}[\mathbf{e}_y \mid I] = \eta(I)$ and $\mathbb{E}[\mathbf{e}_y \mid \varnothing] = \bar{\eta}$. Substituting these outputs into the one-step prediction in Eq. (15), for a general ω we obtain

$$\begin{aligned} z^{(1)}_{\text{Drift}} &= z_0 + (1-\omega)(\bar{\eta} - z_0) + \omega(\eta(I) - z_0) \\ &= \omega\eta(I) + (1-\omega)\bar{\eta}, \\ z^{(1)}_{\text{Leap}} &= (1-\omega)\bar{\eta} + \omega\eta(I). \end{aligned} \tag{26}$$

Setting $\omega = 1$, both parameterizations satisfy

$$z^{(1)}_m = \eta(I), \qquad [z^{(1)}_m]_j = P(y = j \mid I), \qquad m \in \{\text{Drift}, \text{Leap}\},$$

independently of the initial state z_0.

By the decoding rule in Eq. (16), the predicted class is

$$y^m_{\text{pred}} \in \arg\max_j [z^{(1)}_m]_j = \arg\max_j P(y = j \mid I).$$

This also agrees with nearest-neighbor label decoding, since, for any $z \in \mathbb{R}^C$,

$$\|z - \mathbf{e}_j\|_2^2 = \|z\|_2^2 + 1 - 2z_j \quad \Longrightarrow \quad \arg\min_j \|z - \mathbf{e}_j\|_2^2 = \arg\max_j z_j.$$

Given I, predicting class j has conditional error probability $1 - \eta_j(I)$. Choosing a class with maximum posterior probability therefore minimizes the conditional error for each image. For any classifier $g(I, z_0)$ that may depend on the independent initial state,

$$\begin{aligned} P\big(g(I, z_0) \neq y\big) &= \mathbb{E}_{I, z_0}\big[1 - \eta_{g(I,z_0)}(I)\big] \\ &\geq \mathbb{E}_I\big[1 - \max_j \eta_j(I)\big] = P\big(y^m_{\text{pred}} \neq y\big). \end{aligned}$$

When multiple classes share the maximum posterior probability, any fixed measurable tie-breaking rule attains the same optimal risk. This establishes the Bayes-optimal one-step decoding in Corollary 4.4.

Likewise, the Brier risk relationship in the main text for $t = 0$ and $p_{\text{drop}} = 0$ follows directly from

$$\mathbb{E}\big[\|\hat{z}_m(z_0 \mid I) - \mathbf{e}_y\|_2^2 \mid I, z_0\big] = \|\hat{z}_m(z_0 \mid I) - \eta(I)\|_2^2 + 1 - \|\eta(I)\|_2^2$$

The second term on the right-hand side is independent of the predictor, so the unique minimizer of the squared risk is $\eta(I)$.

To verify the ranking statement for general $\omega > 0$, let $p(I) = P(y = 2 \mid I)$, $\bar{p} = P(y = 2)$, and $\mathbf{a} = \mathbf{e}_2 - \mathbf{e}_1$. By Eq. (26),

$$\begin{aligned} \mathbf{a}^\top z^{(1)}_m &= \omega\big(2p(I) - 1\big) + (1-\omega)(2\bar{p} - 1) \\ &= 2\omega p(I) + \big((1-\omega)(2\bar{p} - 1) - \omega\big). \end{aligned}$$

The term in parentheses is image-independent, and $2\omega > 0$. Thus, this score preserves the ordering and ties induced by the posterior probability. When $0 < \bar{p} < 1$, let f_1 and f_2 be the class-conditional densities with respect to a common dominating measure μ. Then

$$p(I) = \frac{\bar{p} f_2(I)}{(1-\bar{p}) f_1(I) + \bar{p} f_2(I)}.$$

Posterior-threshold tests are therefore equivalent to likelihood-ratio-threshold tests. Specifically, let δ^* equal 1 when $f_2 > k f_1$ and 0 when $f_2 < k f_1$, with randomization on equality as needed to

achieve a specified false-positive rate, where $k \geq 0$. For any test δ whose false-positive rate does not exceed that of δ^*,

$$\int (\delta^* - \delta)(f_2 - k f_1)\, d\mu \geq 0,$$

$$\int (\delta^* - \delta) f_2\, d\mu \geq k \int (\delta^* - \delta) f_1\, d\mu \geq 0.$$

Hence, the posterior-threshold test achieves the highest detection rate at that false-positive rate; endpoint levels are handled by limits or by directly choosing the acceptance region. Since positive guidance scales preserve posterior ranking, the Bayes-optimal ranking statement in the main text follows. □

A.3 Proof of Lemma 4.6

Proof. Following the definition of Initial-State Fluctuation, fix the network parameters and image I. All expectations, variances, and covariances are taken over z_0, and the corresponding second moments are assumed finite. The conditional and unconditional branches for the same image use the same initial state. By Eq. (15), both parameterizations satisfy

$$z_m^{(1)} = (1-\omega)\hat{z}_m(z_0 \mid \varnothing) + \omega \hat{z}_m(z_0 \mid I).$$

For FakeFlow-Leap, this follows directly from the linear combination of the branch outputs. For FakeFlow-Drift, the two coefficients sum to 1, giving

$$\begin{aligned} &z_0 + (1-\omega)\mathcal{F}_{\theta_m}(z_0, 0 \mid \varnothing) + \omega \mathcal{F}_{\theta_m}(z_0, 0 \mid I) \\ &\quad = (1-\omega)\big(z_0 + \mathcal{F}_{\theta_m}(z_0, 0 \mid \varnothing)\big) + \omega\big(z_0 + \mathcal{F}_{\theta_m}(z_0, 0 \mid I)\big), \end{aligned}$$

Thus, the same expression holds for Drift. Multiplying by $\mathbf{a}^\top$, using the definitions of s_{c} and $s_\varnothing$, and dividing by $\omega > 0$ yields

$$\frac{s_\omega}{\omega} = s_{\mathrm{c}} - \lambda s_\varnothing, \qquad \lambda = 1 - \omega^{-1}. \tag{27}$$

Let $\mu_\varnothing = \mathbb{E}[s_\varnothing]$. The unconditional branch does not depend on the image, and the initial-state distribution is identical across images. Therefore, $\mu_\varnothing$ is an image-independent constant. By Eq. (27),

$$\mathbb{E}\left[\frac{s_\omega(I)}{\omega}\right] = \mu_{\mathrm{c}}(I) - \lambda \mu_\varnothing,$$

Consequently, for any images I and I',

$$\mathbb{E}\left[\frac{s_\omega(I)}{\omega}\right] - \mathbb{E}\left[\frac{s_\omega(I')}{\omega}\right] = \mu_{\mathrm{c}}(I) - \mu_{\mathrm{c}}(I').$$

This proves the preservation of inter-image mean differences stated in the lemma. Centering the unconditional score gives $s_{\mathrm{c}} - \lambda(s_\varnothing - \mu_\varnothing)$, whose expectation is $\mu_{\mathrm{c}}(I)$. It differs from s_ω/ω only by the constant $\lambda\mu_\varnothing$ and therefore has the same variance. This establishes the corresponding control-variate form.

Subtracting the respective means of the two branches in Eq. (27), expanding the square, and taking expectations gives

$$\begin{aligned} V(\omega) &= \mathbb{E}\Big[\big((s_{\mathrm{c}} - \mu_{\mathrm{c}}) - \lambda(s_\varnothing - \mu_\varnothing)\big)^2\Big] \\ &= \mathbb{E}[(s_{\mathrm{c}} - \mu_{\mathrm{c}})^2] - 2\lambda \mathbb{E}[(s_{\mathrm{c}} - \mu_{\mathrm{c}})(s_\varnothing - \mu_\varnothing)] + \lambda^2 \mathbb{E}[(s_\varnothing - \mu_\varnothing)^2] \\ &= \sigma_{\mathrm{c}}^2 - 2\lambda\kappa + \lambda^2 \sigma_\varnothing^2, \end{aligned}$$

This is exactly the variance formula in Eq. (18).

Since $\omega = 1$ corresponds to $\lambda = 0$, we have $V(1) = \sigma_{\mathrm{c}}^2$. Their difference is therefore

$$V(\omega) - V(1) = \lambda\big(\lambda \sigma_\varnothing^2 - 2\kappa\big). \tag{28}$$

Under the condition $\kappa > 0$, the inequality $\kappa^2 \le \sigma_c^2 \sigma_\varnothing^2$ implies $\sigma_\varnothing^2 > 0$. The map $\omega \mapsto \lambda = 1 - \omega^{-1}$ is strictly increasing and maps $(1, \infty)$ onto $(0, 1)$. Hence, whenever

$$0 < \lambda < \min\left\{1, \frac{2\kappa}{\sigma_\varnothing^2}\right\},$$

Eq. (28) is strictly negative. Returning to the guidance scale, define

$$\bar{\omega} = \begin{cases} \dfrac{\sigma_\varnothing^2}{\sigma_\varnothing^2 - 2\kappa}, & 2\kappa < \sigma_\varnothing^2, \\ +\infty, & 2\kappa \ge \sigma_\varnothing^2, \end{cases}$$

Then $\bar{\omega} > 1$, and $V(\omega) < V(1)$ for every $\omega \in (1, \bar{\omega})$.

If additionally $\kappa < \sigma_\varnothing^2$, completing the square yields

$$V(\omega) = \sigma_\varnothing^2 \left(\lambda - \frac{\kappa}{\sigma_\varnothing^2}\right)^2 + \sigma_c^2 - \frac{\kappa^2}{\sigma_\varnothing^2}.$$

The unique minimizer is then $\lambda^* = \kappa/\sigma_\varnothing^2 \in (0, 1)$, corresponding to

$$\omega^* = \frac{1}{1 - \lambda^*} = \frac{\sigma_\varnothing^2}{\sigma_\varnothing^2 - \kappa} > 1.$$

Using $\rho = \kappa/(\sigma_c \sigma_\varnothing)$, the variance at this point is

$$V(\omega^*) = \sigma_c^2 - \frac{\kappa^2}{\sigma_\varnothing^2} = \left(1 - \frac{\kappa^2}{\sigma_c^2 \sigma_\varnothing^2}\right) \sigma_c^2 = (1 - \rho^2)\sigma_c^2.$$

This proves the optimal guidance scale and minimum variance stated in the lemma.

For the image-dependent case mentioned in the main text, retaining $\sigma_c^2(I)$ and $\kappa(I)$ gives, by the same calculation for each image,

$$V_I(\omega) = \sigma_c^2(I) - 2\lambda\kappa(I) + \lambda^2 \sigma_\varnothing^2.$$

If $\kappa_{\min} = \inf_I \kappa(I) > 0$ over the image set under consideration, replacing κ by $\kappa_{\min}$ in the expression for $\bar{\omega}$ gives a common interval of strict variance reduction for all images. □

A.4 Proof of Corollary 4.7

Proof. By Lemma 4.6, the inter-image mean difference of the normalized scores does not depend on ω. For the fixed image pair (I_f, I_r) in the corollary, define

$$D_\omega = \frac{s_\omega(I_f)}{\omega} - \frac{s_\omega(I_r)}{\omega}.$$

By Eq. (27), the identical unconditional means cancel between the two images:

$$\mathbb{E}[D_\omega] = \mu_c(I_f) - \lambda\mu_\varnothing - \big(\mu_c(I_r) - \lambda\mu_\varnothing\big) = \Delta > 0.$$

The initial states of the two images are sampled independently. Moreover, the main text assumes that σ_c^2 and κ are image-independent. Thus, the two scores are independent, each with variance $V(\omega)$, and

$$\mathrm{Var}(D_\omega) = \mathrm{Var}\left(\frac{s_\omega(I_f)}{\omega}\right) + \mathrm{Var}\left(\frac{s_\omega(I_r)}{\omega}\right) = 2V(\omega).$$

Let $X = D_\omega - \Delta$. Then $\mathbb{E}[X] = 0$ and $\mathbb{E}[X^2] = 2V(\omega)$. For any $b \ge 0$, the event $X \le -\Delta$ implies $(X - b)^2 \ge (\Delta + b)^2$. Applying Markov's inequality to the nonnegative random variable gives

$$\begin{aligned} P(D_\omega \le 0) &= P(X \le -\Delta) \\ &\le \frac{\mathbb{E}[(X - b)^2]}{(\Delta + b)^2} = \frac{2V(\omega) + b^2}{(\Delta + b)^2}. \end{aligned}$$

Choosing $b = 2V(\omega)/\Delta$ yields the Cantelli bound

$$P(D_\omega \leq 0) \leq \frac{2V(\omega)}{2V(\omega) + \Delta^2}.$$

Since $\omega > 0$, the events $s_\omega(I_f) \leq s_\omega(I_r)$ and $D_\omega \leq 0$ are equivalent. Therefore,

$$P\big(s_\omega(I_f) \leq s_\omega(I_r)\big) \leq \frac{2V(\omega)}{2V(\omega) + \Delta^2}, \tag{29}$$

which is the inequality in Corollary 4.7.

Finally, $\Delta > 0$ is independent of ω, and

$$\frac{d}{dV}\frac{2V}{2V + \Delta^2} = \frac{2\Delta^2}{(2V + \Delta^2)^2} > 0.$$

By Lemma 4.6, $V(\omega) < V(1)$ when $\omega \in (1, \bar{\omega})$. Hence,

$$\frac{2V(\omega)}{2V(\omega) + \Delta^2} < \frac{2V(1)}{2V(1) + \Delta^2},$$

establishing a strictly tighter upper bound on the mis-ranking probability within this interval. □