THEORETICALLY PRINCIPLED BALANCED DEEPFAKE DETECTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Driven by advances in deep neural networks (DNNs), modern generative AI has enabled highly realistic media synthesis while also giving rise to deepfakes that threaten social trust, making reliable deepfake detection essential. Although many DNN-based deepfake detectors have been developed, they overemphasize overall performance but largely ignore class imbalance, which can severely degrade performance and induce asymmetric accuracy between classes as the imbalance increases. Existing imbalance-aware deepfake detection methods mainly address class imbalance but overlook domain-level imbalance and sample hardness, leading to detectors fail to reliably detect fakes from specific domains in real-world deployments. In this work, we study a new bi-level imbalanced learning problem and propose the first balanced deepfake detection approach that jointly addresses class-level and domain-level imbalance by designing a bi-level balanced loss with a general guarantee of $\mathcal{H}$-consistency and a Bayes consistency result established for a special case. Extensive experiments demonstrate that our method consistently outperforms existing losses and can be used in various DNN-based deepfake detectors to improve balanced performance and further improve overall performance. The code is available at https://anonymous.4open.science/r/bb_loss-2777.

1 INTRODUCTION

Over the past decade, modern generative AI models, including variational autoencoders (VAE Vahdat & Kautz (2020)), generative adversarial networks (GAN Karras et al. (2020)), and diffusion models (DM Dhariwal & Nichol (2021)), have achieved rapid progress, largely driven by advances in deep neural networks (DNNs). These technologies have substantially improved the ability to synthesize highly realistic images and videos, expanding creative possibilities and enhancing user experiences. At the same time, they introduce significant risks of misuse. A representative example is the emergence of "DeepFakes", which enable the generation of highly convincing forged multimedia content Westerlund (2019). The resulting capability to generate persuasive false information that can influence public opinion or damage individual reputations poses serious challenges to democratic processes and social trust. As a result, effective and reliable deepfake detection has become essential for maintaining confidence in digital information ecosystems.

Recently, numerous deepfake detectors based on deep neural networks have been proposed, aiming to capture spatial- and frequency-based artifact characteristics and exploit these learned features to distinguish authentic from manipulated media Lin et al. (2024a). Despite these advances, class imbalance in the training data remains largely unaddressed. In particular, detector performance can degrade substantially as the class imbalance ratio increases, as demonstrated in Lin et al. (2025).

There are only a few works Pu et al. (2022); Krubha et al. (2025) that specifically address class imbalance in deepfake detection by utilizing existing robust loss functions from the machine learning literature. Although these methods report improved performance under class-imbalanced settings, they **overlook** a *critical* and *distinct* form of imbalance. Specifically, they treat all fake samples as a single homogeneous category, without accounting for the fact that fake content may originate from diverse source domains or generators. As a result, even when balance is achieved between real and fake classes, substantial performance disparities can still arise across domains due to *domain-level imbalance* that is typically ignored during training. In practice, such imbalance naturally arises

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.

from data collection constraints, as certain commercial generators (e.g., $\sim$\$0.06 per image for Nano Banana 2 Google (2026)) are costly to query, which further limits the number of samples that can be generated. This limitation can result in detectors that exhibit strong overall performance yet fail to reliably identify fakes generated by domains that are underrepresented during training but may become prevalent at test time, thereby substantially limiting their reliability in practical deployments. In addition, sample hardness is frequently **overlooked** in existing approaches, which can further undermine balanced performance, as models tend to prioritize learning from easy examples and consequently lazy to learn hard examples Zhou & Wu (2023) that could originate from specific domains. More discussion about the related works can be found in the Appendix.

Note that this constitutes a **new** imbalanced learning problem that differs fundamentally from traditional single-level (e.g., class or domain) imbalance settings in machine learning. Here, we target a **bi-level** imbalanced learning setting in which class-level imbalance **must** be addressed jointly with domain-level imbalance, while mitigating the influence of sample hardness, where domains are often correlated with class labels. As a result, solutions with both empirical effectiveness and theoretical guarantees are required.

To this end, we propose the **first** imbalanced learning approach that simultaneously addresses class-level and domain-level imbalance in deepfake detection. We assume domain labels are available during training but not required at test time, which is standard in existing deepfake datasets (e.g., FaceForensics++ Rossler et al. (2019), AI-Face Lin et al. (2025)), where manipulation types or generators are explicitly annotated and can be obtained during data collection. Specifically, we develop a **B**i-level **B**alanced (**BB**) loss by shifting model logits according to both class and domain priors. To mitigate the influence of sample hardness on imbalanced learning, we further encourage the model to focus on hard examples during training by replacing the conventional average operator with an average top-k operator when aggregating bi-level balanced sample losses. Finally, we provide a comprehensive theoretical analysis of the proposed loss, establishing that it is Bayes-consistent Steinwart (2007) for a special case and also satisfies $\mathcal{H}$-consistent Awasthi et al. (2022) for complete hypothesis classes. Our key contributions are:

- We present a **new** imbalanced learning **problem** arising in deepfake detection.
- We introduce the **first** theoretically guaranteed approach for bi-level balanced deepfake detection.
- Our approach can be **readily applied** in practice by simply replacing BCE with the proposed loss.
- Extensive experiments show that our method **surpasses** state-of-the art approaches and can be **seamlessly integrated** into existing deepfake detectors.

2 Motivation

To illustrate the domain-level imbalance issue, we conduct experiments on the widely used image-based FaceForensics++ (FF++) dataset Rossler et al. (2019), which consists of one real domain and five fake domains corresponding to manipulation methods including DeepFakes DeepFakes (2017), Face2Face Thies et al. (2016), FaceSwap Kowalski (2018), NeuralTextures Thies et al. (2019), and FaceShifter Li et al. (2019). The dataset is randomly divided into training and testing sets. From the training split, we construct additional training sets by randomly sampling data to enforce different real-to-fake ratios (1:1, 10:1, and 100:1), thereby simulating varying degrees of class/domain imbalance.

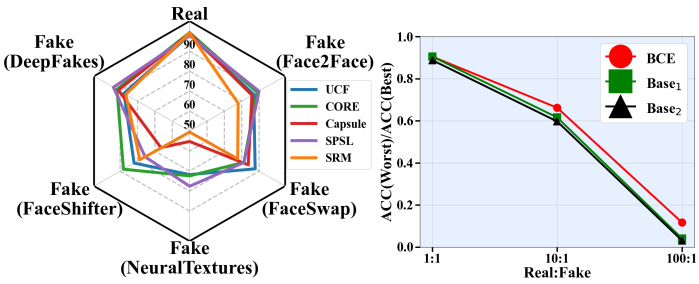

Figure 1: **Left**: Domain-level accuracy (ACC) imbalance exists in various deepfake detectors. **Right**: Class-level imbalance training strategies ($Base_1$ and $Base_2$) are insufficient to mitigate domain-level imbalance.

In the first experiment, we train several state-of-the-art deepfake detectors (i.e., UCF Yan et al. (2023), CORE Ni et al. (2022), Capsule Nguyen et al. (2019), SPSL Liu et al. (2021), and SRM Luo et al. (2021)) on imbalanced training data with a 10:1 real-to-fake ratio and report their domain-level

test performance in Fig. (1) Left. The results show that all methods exhibit pronounced domain-level bias during testing, which motivates our work.

In the second experiment, we train the same detectors using a Xception Chollet (2017) backbone and different loss functions, including the standard **BCE** loss and AUC-based imbalance learning objectives adopted in prior deepfake detection studies (**Base**$_1$ Pu et al. (2022) and **Base**$_2$ Krubha et al. (2025)). Models are trained on newly curated training sets with varying real-to-fake ratios and evaluated on the same fixed test set. To quantify domain-level imbalance, we report the ratio of the worst to the best accuracy (ACC) across domains (higher is better). As shown in Fig. (1) Right, the domain-level test performance of all methods is highly sensitive to the training imbalance ratio. Notably, the imbalance-aware state-of-the-art methods (**Base**$_1$ and **Base**$_2$) perform worse than the BCE baseline, indicating that class-level imbalance remedies alone are insufficient to address domain-level imbalance.

3 METHODOLOGY

Problem Setup. Let $\mathcal{X}$ denote the input space, and let $\mathcal{Y} = \{\text{real}, \text{fake}\}$ and $\mathcal{G} = \{\text{real}, \text{fake}_{d_1}, \ldots, \text{fake}_{d_l}\}$ denote the sets of class labels and domains, respectively, consisting of one real domain and l fake domains. Given a training dataset $\mathcal{D} = \{(x_i, y_i, g_i)\}_{i=1}^n$ of size n, each sample $(x, y, g) \in \mathcal{X} \times \mathcal{Y} \times \mathcal{G}$. We consider a score function $h_\theta : \mathcal{X} \times \mathcal{Y} \times \mathcal{G} \to \mathbb{R}$ drawn from a function set $\mathcal{H}$ and parameterized by θ. When the context is clear, we omit the subscript and write h. The predicted label is defined as $\hat{y}(x) = \arg\max_{y \in \mathcal{Y}} h(x, y, g)$. Let $p(y)$ and $p(g)$ denote the marginal probabilities of class label y and domain g, respectively. Both of them can be estimated from the training set in practice. Our goal is to study a bi-level balanced classification problem, where the objective is to train a deepfake detector based on $\mathcal{D}$ that achieves balanced performance at both the class level and the domain level.

3.1 (SINGLE-LEVEL) BALANCED LOSS

Class imbalance occurs when the label distribution $p(y)$ is highly skewed, such that certain classes (commonly referred to as "tail" classes) appear far less frequently than others (the "head" or majority classes). To enable imbalanced learning, balanced loss (denoted by $\ell^{\mathcal{Y}}_{BAL}$) has been widely adopted in the literature Menon et al. (2013; 2021); Cortes et al. (2025), as they counteract class imbalance by reweighting the loss of each training example according to the inverse of the marginal probability of its true class $p(y)$: $\ell^{\mathcal{Y}}_{BAL}(h, x, y, g) = \mathbf{1}_{\hat{y}(x) \neq y}/p(y)$, where $\mathbf{1}_{[a]}$ denotes the indicator function, taking value 1 if the condition a holds and 0 otherwise. Accordingly, the corresponding population risk can be expressed as $\mathcal{R}_{\ell^{\mathcal{Y}}_{BAL}}(h) = \mathbb{E}[\ell^{\mathcal{Y}}_{BAL}(h, x, y, g)] = \mathbb{E}_y\big[\frac{1}{p(y)}\mathbb{E}_{x,g|y}[\mathbf{1}_{\hat{y}(x) \neq y}]\big]$. To maintain consistency with the notation used below, we include g in the above equations, although it can be omitted. Although minimizing the above risk can achieve balanced performance at the label level, it suffers from two key limitations: **First**, it fails to simultaneously achieve balanced classification performance across *domains*. **Second**, the class-level weighting scheme only addresses class imbalance but overlooks *sample importance*. In practice, some samples are substantially harder to learn than others, which can further induce imbalanced performance during training. These motivate the design of a new learning objective for imbalanced learning.

3.2 BI-LEVEL BALANCED LOSS

Inspired by the class-level weighting scheme used in $\mathcal{R}_{\ell^{\mathcal{Y}}_{BAL}}(h)$, we propose a new risk $\mathcal{R}_{\ell^{\mathcal{G}}_{BAL}}(h)$ to address imbalance at the domain level:

$$\mathcal{R}_{\ell^{\mathcal{G}}_{BAL}}(h) := \mathbb{E}[\ell^{\mathcal{G}}_{BAL}(h, x, y, g)] = \mathbb{E}_g\big[\frac{1}{p(g)}\mathbb{E}_{x,y|g}[\mathbf{1}_{\hat{y}(x) \neq y}]\big], \tag{1}$$

where the loss at the individual sample level is defined as $\ell^{\mathcal{G}}_{BAL}(h, x, y, g) := \mathbf{1}_{\hat{y}(x) \neq y}/p(g)$. The indicator function $\mathbf{1}_{\hat{y}(x) \neq y}$ in Eq. equation 1 indicates that $\mathcal{R}_{\ell^{\mathcal{G}}_{BAL}}(h)$ focuses on label classification performance. By reweighting each sample by the inverse marginal probability of its domain g, this risk effectively compensates for domain-level imbalance.

Formulation. Our goal is to minimize the label-level balanced risk $\mathcal{R}_{\ell^{\mathcal{Y}}_{BAL}}$ while simultaneously controlling the domain-level balanced risk $\mathcal{R}_{\ell^{\mathcal{G}}_{BAL}}$. This leads to the following constrained optimization problem: $\min_h \mathcal{R}_{\ell^{\mathcal{Y}}_{BAL}}(h)$, s.t. $\mathcal{R}_{\ell^{\mathcal{G}}_{BAL}}(h) \leq \epsilon$, where $\epsilon \geq 0$ is a predefined upper bound on the domain-level risk. This constrained problem can be reformulated as the following unconstrained optimization problem:

$$\min_h \mathcal{R}_{\ell^{\mathcal{Y}}_{BAL}}(h) + \beta \mathcal{R}_{\ell^{\mathcal{G}}_{BAL}}(h), \tag{2}$$

where $\beta > 0$ is a trade-off hyperparameter that balances class-level accuracy and domain-level accuracy. By substituting the definitions of the risks, Eq. (2) can be rewritten as

$$\min_h \mathbb{E}\Big[\underbrace{\frac{\mathbf{1}_{\hat{y}(x)\neq y}}{p(y)}}_{①} + \beta \underbrace{\frac{\mathbf{1}_{\hat{y}(x)\neq y}}{p(g)}}_{②} \Big]. \tag{3}$$

It is evident that the learning objective contains two balanced loss terms, forming a Bi-Level Balanced (**BB**) Loss. Specifically, ① aims to balance *class-level* performance in the presence of class imbalance, while ② is designed to balance *domain-level* label classification performance.

Relaxation. Since the 0–1 loss in Eq. (3) is non-differentiable and thus unsuitable for gradient-based optimization, numerous surrogate losses have been proposed to replace the balanced loss Byrd & Lipton (2019); Menon et al. (2021). However, as discussed in Cortes et al. (2025), many of these surrogates are not Bayes-consistent, a property that is crucial for ensuring that *minimizing the surrogate loss is equivalent to minimizing the original 0-1 based loss*. For this reason, we adopt the Bayes-consistent Generalized Logit-Adjusted (GLA) loss Cortes et al. (2025) as a surrogate for the balanced loss. In particular, the term ① can be replaced by

$$\Psi_{\mathcal{Y}}(h,x,y,g) := \Phi^q\left(\frac{e^{h(x,y,g)+\frac{\log(p(y))}{1-q}}}{\sum_{y'\in\mathcal{Y}} e^{h(x,y',g)+\frac{\log(p(y'))}{1-q}}} \right), \tag{4}$$

where $\Phi^q(t) = -\log(t)$ if $q = 0$ and $\frac{1}{q}(1-t^q)$ if $q \in (0,\infty)$.

Indeed, replacing the term ② is non-trivial. To this end, we rewrite ② to $\frac{\mathbf{1}_{\hat{y}(x)\neq y}}{\mathbf{1}_{\hat{g}(x)\neq g}} \cdot \frac{\mathbf{1}_{\hat{g}(x)\neq g}}{p(g)}$, where the denominator $\mathbf{1}_{\hat{g}(x)\neq g}$ is nonzero (this condition can be removed in the subsequent analysis). Here, $\hat{g}(x) = \arg\max_{g\in\mathcal{G}} h^o(x,y,g)$ denotes the predicted domain, where the auxiliary score function $h^o : \mathcal{X}\times\mathcal{Y}\times\mathcal{G} \to \mathbb{R}$ is also drawn from the function set $\mathcal{H}$. Following previous relaxation for ①, the second term in this decomposition, $\frac{\mathbf{1}_{\hat{g}(x)\neq g}}{p(g)}$, can then be replaced by $\Phi^q\left(\frac{e^{h^o(x,y,g)+\frac{\log(p(g))}{1-q}}}{\sum_{g'\in\mathcal{G}} e^{h^o(x,y,g')+\frac{\log(p(g'))}{1-q}}} \right)$. Therefore, ② can be replace by

$$\frac{\mathbf{1}_{\hat{y}(x)\neq y}}{\mathbf{1}_{\hat{g}(x)\neq g}} \cdot \Phi^q\left(\frac{e^{h^o(x,y,g)+\frac{\log(p(g))}{1-q}}}{\sum_{g'\in\mathcal{G}} e^{h^o(x,y,g')+\frac{\log(p(g'))}{1-q}}} \right). \tag{5}$$

However, the format of Eq. (5) has two issues. **First**, the term $\frac{\mathbf{1}_{\hat{y}(x)\neq y}}{\mathbf{1}_{\hat{g}(x)\neq g}}$ is non-differentiable. **Second**, and more critically, this formulation requires learning an additional score function h^o for domain prediction, which would increase the model complexity. Fortunately, we have the following theorem:

Theorem 1. *(5) is equivalent to* $\Psi_{\mathcal{G}}(h,x,y,g) := \Phi^q\Bigg(\underbrace{\frac{e^{h(x,y(g),g)+\frac{\log(p(g))}{1-q}}}{\sum_{g'\in\mathcal{G}} e^{h(x,y(g'),g')+\frac{\log(p(g'))}{1-q}}}}_{③} \Bigg)$, *where* $y(g)$ *is a label–domain matching function that depends on the domain, defined as* $y(g) =$ *real if* $g =$ *real and* $y(g) =$ *fake if* $g \in \{\text{fake}_{d_1}, ..., \text{fake}_{d_l}\}$.

The proof of Theorem 1 can be found in the Appendix. The domain-balancing behavior of ③ can be further clarified. Specifically, we rewrite ③ as $\frac{1}{\sum_{g'\in\mathcal{G}} e^{h(x,y(g'),g')-h(x,y(g),g)} \cdot \left(p(g')/p(g)\right)^{\frac{1}{1-q}}}$. The

factor $\big(p(g')/p(g)\big)^{\frac{1}{1-q}}$ in the denominator serves as a domain-dependent weighting term, inducing a pairwise margin adjustment based on the relative frequencies of domain g and other domains g'. As a result, the margin is enlarged when g corresponds to a rare domain with low $p(g)$ and g' is a frequent domain with high $p(g')$, thereby preventing scores from dominant domains from overwhelming those of underrepresented domains.

By substituting $\Psi_{\mathcal{Y}}(h, x, y, g)$ and $\Psi_{\mathcal{G}}(h, x, y, g)$ for ① and ②, respectively, we obtain the final relaxed formulation:

$$\min_h \mathbb{E}[\Psi(h, x, y, g)], \tag{6}$$

where $\Psi(h, x, y, g) := \Psi_{\mathcal{Y}}(h, x, y, g) + \beta\Psi_{\mathcal{G}}(h, x, y, g)$.

3.3 Learning Objective with Robust Aggregation

As is standard in machine learning, rather than minimizing the unknown population risk in Eq. (6), we optimize its empirical counterpart, given by the objective $\frac{1}{n}\sum_{i=1}^{n}\Psi(h, x_i, y_i, g_i)$. However, this formulation overlooks sample hardness, as some training examples are inherently easier to learn than others Zhou & Wu (2023), which can adversely affect balanced performance and generalization. To mitigate this issue, we replace the conventional average operator with an average top-k aggregator Hu et al. (2023), encouraging the training process to focus on harder examples. The resulting learning objective is

$$\min_h \frac{1}{k}\sum_{i=1}^{k}\Psi_{[i]}(h, x, y, g), \tag{7}$$

where $\Psi_{[i]}(h, x, y, g)$ is the i-th largest individual loss in $\{\Psi(h, x_i, y_i, g_i) : i \in \{1, ..., n\}\}$ (ties can be broken in any consistent way) and $k \in \{1, ..., n\}$ is a hyperparameter.

Reformulation. To avoid an explicit sorting procedure during training, we exploit an equivalent formulation based on the following result.

Proposition 1. *(Lemma 1 of Hu et al. (2021)) For a set of real numbers* $\overline{\Psi} = \{\Psi(h, x_1, y_1, g_1), ..., \Psi(h, x_n, y_n, g_n)\}$*, let* $\Psi_{[k]}(h, x, y, g)$ *denote the* k*-th largest value in* $\overline{\Psi}$ *for* $k \in \{1, ..., n\}$*. Then we have* $\frac{1}{k}\sum_{i=1}^{k}\Psi_{[i]}(h, x, y, g) = \min_{\lambda\in\mathbb{R}}\{\lambda + \frac{1}{k}\sum_{i=1}^{n}[\Psi(h, x_i, y_i, g_i) - \lambda]_+\}$*, where* $[a]_+ = \max\{0, a\}$ *is the hinge function.*

For completeness, we provide its proof in the Appendix. To avoid searching over a large range of values for k, we set $k = \alpha n$ with $\alpha \in (0, 1]$, which can be tuned on a predefined grid in practice. Using the above proposition, we derive the final learning objective by reformulating Eq. (7) as

$$\min_{h,\lambda} \lambda + \frac{1}{\alpha n}\sum_{i=1}^{n}[\Psi(h, x_i, y_i, g_i) - \lambda]_+, \tag{8}$$

It is evident that Eq. (8) introduces an additional robust sample loss aggregation mechanism to strengthen imbalanced learning. More importantly, the proposed formulation in Eq. (8) is highly general. Specifically, when $\beta = 0$ and $\alpha = 1$, the formulation reduces to the Generalized Logit-Adjusted learning objective Cortes et al. (2025). As such, it extends the concept of logit adjustment to a broader family of generalized cross-entropy losses Mao et al. (2023). When $q = 0$, $\beta = 0$, and $\alpha = 1$, the formulation reduces to the traditional Logit-Adjusted learning objective Menon et al. (2021).

Optimization. The optimization problem in Eq. (8) can be solved using an iterative stochastic gradient descent procedure. In practice, we initialize the model parameters θ of h and repeatedly sample a mini-batch $\mathcal{D}_b$ of size $|\mathcal{D}_b|$ from set $\mathcal{D}$, performing the following two steps at each iteration t on $\mathcal{D}_b$: (1) We fix θ and use binary search to find the global optimum of λ since (8) is convex w.r.t. λ. (2) We fix λ and update θ using (stochastic) gradient descent with a user-specified learning rate $\eta > 0$: $\theta_{t+1} = \theta_t - \frac{\eta}{\alpha|\mathcal{D}_b|}\sum_{i\in\mathcal{D}_b}\partial_\theta\Psi(h, x_i, y_i, g_i)\cdot\mathbf{1}_{[\Psi(h,x_i,y_i,g_i)>\lambda]}$, where $\partial_\theta\Psi$ represents the (sub)gradient of Ψ w.r.t. θ. The iteration is terminated once a user-specified stopping criterion is met (e.g., a maximum number of iterations). Note that this optimization process is similar to how one would train the original deepfake detector being modified except that we now have an additional binary search to update λ, resulting in a comparable overall training complexity.

4 THEORETICAL GUARANTEES

In this section, we investigate the statistical property of the proposed loss in Eq. equation 8. In particular, we analyze the Bayes- and $\mathcal{H}$-consistency of its population analogue, $\inf_{\lambda\in\mathbb{R}} \mathbb{E}\left[\lambda + \frac{1}{\alpha}[\Psi(h,x,y,g) - \lambda]_+\right]$.

For clarity, we first introduce some notations for the convenience of our theoretical analysis. We consider a data distribution $\mathcal{D} \subset \mathcal{X}\times\mathcal{Y}\times\mathcal{G}$ and the hypothesis set $\mathcal{H}$ consists of all score functions h. We define the minimum label and domain probabilities as $p^{\mathcal{Y}}_{\min} = \min_{y\in\mathcal{Y}} p(y), p^{\mathcal{G}}_{\min} = \min_{y\in\mathcal{G}} p(g)$ and the overall minimum margin as $p_{\min} = \min\{p^{\mathcal{Y}}_{\min}, p^{\mathcal{G}}_{\min}\}$. For any loss function ℓ, the population risk (generalization error) of the hypothesis h is $\mathcal{R}_\ell(h) = \mathbb{E}_{(x,y,g)\sim\mathcal{D}}\ell(h,x,y,g)$. The minimal $(\ell,\mathcal{H})$-risk is defined as the lowest possible generalization error achievable within the hypothesis set $\mathcal{H}$, i.e. $\mathcal{R}^*_\ell(\mathcal{H}) = \inf_{h\in\mathcal{H}} \mathcal{R}_\ell(h)$.

A surrogate loss ℓ_1 is often employed in place of a target loss ℓ_2, and a key theoretical property of such surrogates is **Bayes-consistency** Steinwart (2007). This property ensures that, for the class of all measurable functions $\mathcal{H}_{\text{all}}$, any sequence of predictors $\{h_n\}_{n\geq 1} \subset \mathcal{H}_{\text{all}}$ that asymptotically minimizes the surrogate risk $\lim_{n\to\infty} \mathcal{R}_{\ell_1}(h_n) = \mathcal{R}^*_{\ell_1}(\mathcal{H}_{\text{all}})$ also achieves the optimal target risk in the limit, i.e., $\lim_{n\to\infty} \mathcal{R}_{\ell_2}(h_n) = \mathcal{R}^*_{\ell_2}(\mathcal{H}_{\text{all}})$. However, it is inherently asymptotic and defined only for $\mathcal{H}_{\text{all}}$. In practical scenarios, learning is often performed over a restricted class $\mathcal{H}$, making **$\mathcal{H}$-consistency** and associated non-asymptotic bounds more relevant. These bounds quantify how minimizing the surrogate risk within $\mathcal{H}$ translates into control over the target risk, offering a more precise understanding of surrogate performance in finite-sample settings Awasthi et al. (2021; 2022).

Our theoretical results are summarized in the following theorems. When $\alpha = 1$, the proposed surrogate loss reduces to Ψ, and Theorem 2 establishes that it is Bayes-consistent with respect to the balanced loss $\Psi_{BAL} := \ell^{\mathcal{Y}}_{BAL} + \beta\ell^{\mathcal{G}}_{BAL}$. However, for $0 < \alpha < 1$, the Bayes-consistency no longer holds. This is because the average top-k aggregation emphasizes worst-case (upper-tail) losses, and a decision that is Bayes-optimal in expectation need not minimize tail risk. We provide an explicit counterexample in Appendix. Furthermore, Theorem 3 shows that the surrogate is also $\mathcal{H}$-consistent with the balanced loss Ψ_{BAL}. The proofs of theorems can be found in the Appendix.

Theorem 2 (Bayes-consistency)**.** *For any $q \in [0,1)$, the proposed loss $\inf_{\lambda\in\mathbb{R}} \mathbb{E}\left[\lambda + \frac{1}{\alpha}[\Psi(h,x,y,g) - \lambda]_+\right]$ with $\alpha = 1$ is Bayes-consistent with respect to the combined balanced loss $\Psi_{BAL} := \ell^{\mathcal{Y}}_{BAL} + \beta\ell^{\mathcal{G}}_{BAL}$.*

Theorem 3 ($\mathcal{H}$-consistency)**.** *Assume that $\mathcal{H}$ is complete and that $0 < \beta \leq 1$. Further, suppose that the approximation error satisfies $\mathcal{A}_\Psi(\mathcal{H}) = 0$. Then, for any $q \in [0,1)$, the loss $\inf_{\lambda\in\mathbb{R}} \mathbb{E}\left[\lambda + \frac{1}{\alpha}[\Psi(h,x,y,g) - \lambda]_+\right]$ is $\mathcal{H}$-consistent with respect to the balanced loss Ψ_{BAL}. More precisely, the following $\mathcal{H}$-consistency bound holds:*

$$\widetilde{\mathcal{R}}_{\Psi_{BAL}}(h) - \widetilde{\mathcal{R}}^*_{\Psi_{BAL}}(\mathcal{H}) \leq \widetilde{\Gamma}(\mathcal{R}_\Psi(h) - \mathcal{R}^*_\Psi(\mathcal{H})),$$

where $\widetilde{\Gamma}(t) = \sqrt{2t}/(\alpha\, p_{\min})$ for $q = 0$ and $\widetilde{\Gamma}(t) = \sqrt{2t}/(\alpha\,(p_{\min})^{\frac{1}{1-q}}(1-q)^{1/2})$ for $q \in (0,1)$. Here $\widetilde{\mathcal{R}}_{\Psi_{BAL}}(h)$ denotes the population risk of Ψ_{BAL} under the average top-k aggregator: $\widetilde{\mathcal{R}}_{\Psi_{BAL}}(h) = \inf_{\lambda\in\mathbb{R}} \mathbb{E}[\lambda + \frac{1}{\alpha}[\Psi_{BAL}(h,x,y,g) - \lambda]_+]$, and $\widetilde{\mathcal{R}}^*_{\Psi_{BAL}}(\mathcal{H}) = \inf_{h\in\mathcal{H}} \widetilde{\mathcal{R}}_{\Psi_{BAL}}(h)$ is the corresponding minimal $(\ell,\mathcal{H})$-risk.

Remarks. The approximation error $\mathcal{A}_\ell(\mathcal{H}) = \mathbb{R}^*(\mathcal{H}) - \mathbb{R}^*(\mathcal{H}_{\text{all}})$ quantifies the limitation of the hypothesis class $\mathcal{H}$ in approximating the true target. We assume $\mathcal{A}_\ell(\mathcal{H}) = 0$ in Theorem 3, i.e., $\mathcal{H}$ is rich enough to contain a Bayes-optimal predictor. This is standard in consistency analysis and holds, e.g., when $\mathcal{H}$ is a universal function class or contains the true parametric form of the target.

5 EXPERIMENTS

5.1 SETTINGS

Datasets. We train and evaluate models using different methods on various image-based deepfake datasets: FF++ Rossler et al. (2019), DFD Research (2019), Celeb-DFLi et al. (2020), DFDC Dolhansky et al. (2020), and AI-Face Lin et al. (2025). The FF++ dataset contains one real domain and five fake domains, while AI-Face includes one real domain and three major fake domains (i.e., VAE-generated, GAN-generated, and DM-generated fakes). Each dataset is randomly split into training (80%) and test (20%) sets. Based on the training set, we further simulate class imbalance

Table 1: Intra-dataset performance comparison across different imbalance ratios (i.e., 10:1, 100:1), different datasets, and different backbones. ↑ means higher is better. The best results are shown in **Bold**. The relative improvements between ours and the best from others are shown in red.

Backbone	Real:Fake	Method	FF++ ACC↑	FF++ AUC↑	FF++ Degree_C↑	FF++ Degree_D↑	AI-Face ACC↑	AI-Face AUC↑	AI-Face Degree_C↑	AI-Face Degree_D↑
Xception	10:1	BCE	0.8312	0.9819	0.8033	0.6620	0.9692	0.9981	0.9662	0.8897
		Base_1 Pu et al. (2022)	0.7996	0.9766	0.7658	0.6162	0.9422	0.9975	0.9249	0.7757
		Base_2 Krubha et al. (2025)	0.8358	0.9785	0.8112	0.5972	0.9663	0.9980	0.9631	0.9007
		Ours	**0.9299** (+11.26%)	0.9781	**0.9860** (+21.55%)	**0.8958** (+35.32%)	**0.9692** (+0.00%)	0.9957	**0.9899** (+2.45%)	**0.9026** (+0.21%)
	100:1	BCE	0.4456	0.8835	0.3358	0.1172	0.8225	0.9808	0.7661	0.4377
		Base_1 Pu et al. (2022)	0.3413	0.8019	0.2128	0.0413	0.6464	0.9666	0.5322	0.0015
		Base_2 Krubha et al. (2025)	0.3361	0.8697	0.2005	0.0315	0.8008	0.9806	0.7397	0.5604
		Ours	**0.5526** (+24.01%)	0.8653	**0.4754** (+41.57%)	**0.2754** (+134.98%)	**0.8991** (+9.31%)	0.9797	**0.9048** (+18.10%)	**0.7323** (+30.67%)
EfficientNet-B4	10:1	BCE	0.6370	0.9270	0.5718	0.4285	0.9502	0.9975	0.9373	0.8007
		Base_1 Pu et al. (2022)	0.5178	0.9176	0.4215	0.3124	0.9413	0.9963	0.9252	0.7948
		Base_2 Krubha et al. (2025)	0.6200	0.9416	0.5457	0.3438	0.9608	0.9967	0.9599	0.8957
		Ours	**0.7776** (+22.07%)	0.9091	**0.7862** (+37.49%)	**0.6543** (+52.69%)	**0.9690** (+0.85%)	0.9948	**0.9640** (+0.43%)	**0.9345** (+4.33%)
	100:1	BCE	0.1910	0.7068	0.0245	0.0100	0.6983	0.9638	0.6010	0.2764
		Base_1 Pu et al. (2022)	0.1709	0.6715	0.0000	0.0000	0.5514	0.9558	0.4065	0.0016
		Base_2 Krubha et al. (2025)	0.2141	0.7255	0.0523	0.0177	0.7606	0.9751	0.6841	0.3230
		Ours	**0.3386** (+58.15%)	0.6765	**0.2158** (+312.62%)	**0.1146** (+547.46%)	**0.9011** (+18.47%)	0.9698	**0.9668** (+41.32%)	**0.8324** (+157.71%)

Table 2: Cross-dataset performance. Models are trained on FF++ (100:1) and evaluated on other datasets.

Dataset	DFD			Celeb-DF			DFDC			FF++				AI-Face			
Metrics	ACC	AUC	Degree_C	ACC	AUC	Degree_C	ACC	AUC	Degree_C	ACC	AUC	Degree_C	Degree_D	ACC	AUC	Degree_C	Degree_D
BCE	0.4705	0.7197	0.3698	0.2309	0.6285	0.1205	0.5156	0.5604	0.0334	0.4456	0.8835	0.3358	0.1172	0.4531	0.7060	0.3114	0.1535
Base_1	0.5241	0.7133	0.5072	0.1664	0.5817	0.0389	0.5092	0.5560	0.0131	0.3413	0.8019	0.2128	0.0413	0.3045	0.6643	0.0843	0.0154
Base_2	0.4222	0.7712	0.2902	0.1975	0.6308	0.0788	0.5119	0.5924	0.0261	0.3361	0.8697	0.2005	0.0315	0.3421	0.7532	0.1334	0.0072
Ours	**0.5341**	**0.7214**	**0.4989**	**0.3225**	**0.6024**	**0.2561**	**0.5218**	0.5312	**0.1196**	**0.5526**	0.8653	**0.4754**	**0.2754**	**0.4820**	0.5747	**0.5675**	**0.3855**

by constructing new training sets with ratios 1:100, 1:10, 1:1, 10:1, 100:1 for #Real:#Fake. We do not explicitly resample domain ratios, as the original datasets are already domain-imbalanced, and modifying class imbalance inherently alters the degree of domain imbalance.

Evaluation Metrics. For overall utility, we report accuracy (*ACC*) as the primary performance metric. To quantify imbalance in accuracy across domains or classes, we further compute $\frac{\text{Worst ACC}}{\text{Best ACC}}$ (higher is better), referred to as $\textit{Degree}_C$ and $\textit{Degree}_D$ for class- and domain-level evaluations, respectively. A threshold of 0.5 is used to calculate ACC. *Although our approach is not designed to optimize AUC, we also report it for reference*.

Baselines. To clearly demonstrate the effectiveness of our method, we consider three types of baselines. Baselines in deepfake detection: (1) **Ori**, a backbone model (e.g., Xception Chollet (2017), EfficientNet-B4 Tan & Le (2019)) trained solely with the standard BCE loss; (2) $\textbf{Base}_1$ Pu et al. (2022), which trains the Ori detector by combining BCE loss with an AUC loss; and (3) $\textbf{Base}_2$ Krubha et al. (2025), which trains the Ori detector using a CVaR loss combined with an AUC loss. In addition, to demonstrate the plug-and-play flexibility of our approach, we replace the standard BCE with our proposed learning objective in various deepfake detectors spanning two categories: spatial-based (UCF Yan et al. (2023), CORE Ni et al. (2022), and Capsule Nguyen et al. (2019)) and frequency-based (SPSL Liu et al. (2021) and SRM Luo et al. (2021)). Baselines in general imbalanced learning based on loss design: Cross-Entropy (CE) loss, Logit Adjusted (LA) loss Menon et al. (2021), Equalization (EQUAL) loss Tan et al. (2020), Class-Balanced (CB) loss Cui et al. (2019), FOCAL loss Lin et al. (2018), LDAM loss Cao et al. (2019), and Generalized Logit-Adjusted (GLA) loss Cortes et al. (2025). Baselines in Distributionally Robust Optimization (DRO): Group-DRO Sagawa et al. (2020) and CVaR-DRO Levy et al. (2020).

5.2 Results

Overall Performance. We report the performance of each method in Table 1 across different settings. Our approach consistently outperforms all existing baselines. In particular, it simultaneously mitigates performance bias across both classes and domains without sacrificing overall accuracy. For example, on the FF++ dataset with a 10:1 imbalance ratio, our method achieves relative improvements of 37.49% on Degree_C and 52.69% on Degree_D compared to the strongest baseline. These gains become substantially larger as the imbalance ratio worsens, while also improving overall ACC. Similar trends are observed on the AI-Face dataset. Notably, $\textbf{Base}_1$ and $\textbf{Base}_2$, which are specifically designed to address class imbalance using AUC-based losses, do not exhibit clear improvements in overall ACC or Degree_C compared to BCE. This is because AUC-based objectives primarily optimize ranking performance rather than reducing class-wise 0-1 performance disparities. *Although not designed to improve AUC, our method achieves competitive AUC performance*.

Cross-Dataset Performance. To demonstrate the impact of our method for model generalization, we train models on FF++ and evaluate them on other datasets. For datasets without domain labels, we report only ACC and $\textit{Degree}_C$. As shown in Table 2, our method outperforms the compared methods, although it is not specifically designed to improve cross-dataset generalization.

Table 3: Performance across different detectors.

Type	Method	ACC	Degree_C	Degree_D
Spatial	UCF	0.8566	0.8470	0.7312
	Ours (UCF)	**0.9196** (+7.35%)	**0.9730** (+14.87%)	**0.9035** (+23.56%)
	CORE	0.8693	0.8506	0.7296
	Ours (CORE)	**0.9883** (+13.69%)	**0.9745** (+14.56%)	**0.9694** (+32.86%)
	Capsule	0.8014	0.7738	0.5584
	Ours (Capsule)	**0.8633** (+7.72%)	**0.9764** (+26.18%)	**0.834** (+49.35%)
Frequency	SPSL	0.8612	0.8561	0.7736
	Ours (SPSL)	**0.8755** (+1.66%)	**0.8891** (+3.85%)	**0.8055** (+4.12%)
	SRM	0.7826	0.7474	0.5101
	Ours (SRM)	**0.8683** (+10.95%)	**0.8742** (+16.96%)	**0.7310** (+43.30%)

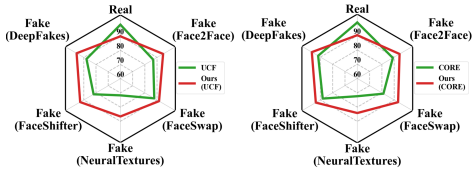

Figure 2: Performance across domains.

Table 4: Performance on modern backbones.

Backbone	Method	ACC↑	AUC	Degree_C↑	Degree_D↑
ConvNeXt-Base	BCE	0.9827	0.9976	0.9396	0.9350
	Base_1	0.9802	0.9973	0.9502	0.9432
	Base_2	0.9842	0.9985	0.9223	0.9208
	GLA	0.9816	0.9973	0.9719	0.9607
	Ours	**0.9864**	0.9981	**0.9837**	**0.9776**
ViT-B/16	BCE	0.9578	0.9901	0.7938	0.7899
	Base_1	0.9484	0.9925	0.7127	0.7116
	Base_2	0.9547	0.9880	0.7956	0.8041
	GLA	0.9336	0.9801	0.6495	0.6474
	Ours	**0.9598**	**0.9926**	**0.9982**	**0.8788**

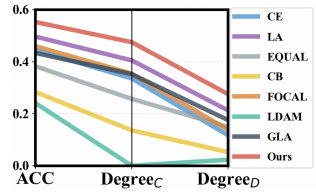

Figure 3: Our method vs. general imbalanced learning baselines.

Plug-and-Play Applicability Across Different Detectors. To demonstrate the broad applicability of our approach, we integrate it into a diverse set of state-of-the-art deepfake detectors spanning both spatial-based and frequency-based categories. As shown in Table 3, regardless of the underlying detection strategy, our method consistently improves the original detectors' ACC and reduces performance bias. These results confirm that our approach is *model-agnostic*. Fig. 2 presents the performance comparison across domains, from which it is evident that detectors trained with our method achieve substantially more balanced domain-level performance.

Generalization to Modern Backbones. To demonstrate that our method is architecture-agnostic and works on top of more recent state-of-the-art backbone networks as well, we also test our BB model with ConvNeXt-Base and ViT-B/16 backbones. As one can see from Table 4, BB also outperforms BCE, Base_1, Base_2, and GLA on all target metrics (ACC, Degree_C, Degree_D).

Comparison with Baselines from General Imbalanced Learning and DRO. We compare our approach with general imbalanced learning baselines under 100:1 ratio setting and report the results in Fig. 3. The results show that single-level, class-oriented imbalanced learning methods perform worse than our bi-level approach. More experiments can be found in Appendix E. We further compare our method with DRO baselines in Table 5 (Left) under both balanced (1:1) and imbalanced (100:1) settings. Although DRO methods focus on minimizing worst-domain risk or hard-sample risk, they do not explicitly address class-level imbalance, and their performance degrades substantially under severe imbalance. In contrast, our method jointly optimizes class-level balance, domain-level balance, and sample importance within a unified framework, leading to the best overall performance across both settings.

5.3 Sensitivity Analysis

Performance under Balanced Settings. To demonstrate the effectiveness of our approach even under balanced sampling settings, we conduct two types of experiments by curating the training data to achieve either class balance or domain balance. Under both settings, we compare our method with existing baselines. As reported in Table 5 (Right), our approach still consistently outperforms the baselines. The reason is that, in practice, it is generally not possible to achieve class balance and domain balance simultaneously through simple resampling or downsampling strategies. Therefore, imbalance cannot be fully eliminated by data curation alone, and our method provides the first principled solution to address this challenge.

Sensitive Analysis on Imbalance Ratios. Fig. 4 (Left) reports the performance of all methods on FF++ with an Xception backbone under various curated imbalance ratios. (1) Our method consistently outperforms all baselines in terms of Degree_C and Degree_D across all ratio settings. Notably, even under the balanced 1:1 ratio, our approach achieves the best performance, highlighting the persistent impact of domain imbalance. By explicitly addressing domain imbalance, our method further alleviates class-level performance bias compared to existing approaches. (2) As the imbalance ratio increases, the degrees of all baseline methods degrades, whereas our approach remains comparatively stable in mitigating performance bias when the ratio varies from 1:1 to 1:100. One possible explanation is that increasing the number of fake samples also enlarges the smallest fake domain, making its size closer to that of the real domain and partially restoring domain balance, a scenario that our method is particularly effective at handling. (3) This can also explain why positive

Table 5: **Left**: Our method vs. DRO-based baselines. **Right**: Performance under balanced settings.

Setting	Method	ACC	Degree_C	Degree_D
1:1	Group-DRO	0.9822	0.9204	0.9176
	CVaR-DRO	0.9838	0.9467	0.9431
	Ours	**0.9844**	**0.9738**	**0.9695**
100:1	Group-DRO	0.3921	0.2744	0.0617
	CVaR-DRO	0.3349	0.1981	0.0927
	Ours	**0.5526**	**0.4754**	**0.2754**

Method	Balanced Class			Balanced Domain		
Metrics	ACC	Degree_C	Degree_D	ACC	Degree_C	Degree_D
BCE	0.9684	0.9142	0.9050	0.9846	0.9413	0.9373
Base_1	0.9339	0.9463	0.9052	0.9769	0.9075	0.9027
Base_2	0.9729	0.8949	0.8885	0.9833	0.9154	0.9134
Ours	**0.9779**	**0.9702**	**0.9603**	**0.9889**	**0.9802**	**0.9761**

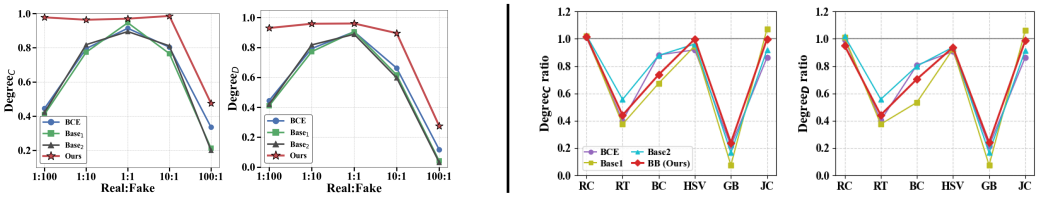

Figure 4: **Left**: Sensitive analysis on imbalance ratios. **Right**: Performance ratio after vs. before post-processing. Points closer to 1.0 (i.e., no post-processing) indicate better robustness.

imbalance (#Real>#Fake) has a more pronounced negative impact on performance than negative imbalance (#Real<#Fake). The reason is that there is only a single real domain, and enlarging it exacerbates both class-level and domain-level imbalance.

Robustness Evaluation under Common Corruptions. We evaluate the influence of our method on model robustness by subjecting the test set to a suite of common image corruptions, including Random Crop (RC), Rotation (RT), Random Brightness Contrast (BC), Hue-Saturation-Value (HSV), Gaussian Blur (GB), and JPEG compression (JC). As illustrated in Fig. 4 (Right), our method follows a similar trend to the baselines and does not introduce additional adverse effects on robustness.

5.4 Ablation Study

Table 6: Ablation study on each loss term and robust aggregation.

Method	Component			Metric ↑			
	$\Psi_{\mathcal{Y}}$	$\Psi_{\mathcal{G}}$	Aggregation	ACC	AUC	Degree_C	Degree_D
Variant1	✓			97.64 ± 0.55	99.43 ± 0.22	95.57 ± 1.06	95.51 ± 1.87
Variant2	✓		✓	97.35 ± 0.87	99.67 ± 0.09	97.13 ± 0.73	93.98 ± 2.12
Variant3		✓		2.62 ± 0.46	0.90 ± 0.11	3.08 ± 1.80	0.42 ± 0.43
Variant4		✓	✓	3.43 ± 1.04	0.75 ± 0.10	1.65 ± 0.97	0.40 ± 0.33
Variant5	✓	✓		98.07 ± 0.25	99.67 ± 0.05	97.39 ± 1.10	95.67 ± 0.21
Ours	✓	✓	✓	$\mathbf{98.37 \pm 0.14}$	$\mathbf{99.75 \pm 0.04}$	$\mathbf{97.95 \pm 0.33}$	$\mathbf{97.23 \pm 0.37}$

To assess the contribution of each loss component (i.e., $\Psi_{\mathcal{Y}}$ and $\Psi_{\mathcal{G}}$) and the role of robust aggregation, we construct four variants of the proposed learning objective and report their test performance trained on the original FF++ training set using an Xception backbone in Table 6. Variant1 corresponds to the latest imbalanced learning objective, GLA Cortes et al. (2025). Comparing Variant1 with Variant2 highlights the importance of accounting for sample hardness during training, showing that robust aggregation further improves performance and reduces bias. Comparing Variant2 with our full model demonstrates the critical role of the domain-level loss term $\Psi_{\mathcal{G}}$ in mitigating domain-level performance imbalance. The comparison between our full model and Variant4 shows that the class-level loss term $\Psi_{\mathcal{Y}}$ is essential for preserving overall detection performance. Variant3 indicates that relying solely on $\Psi_{\mathcal{G}}$ cannot substitute for the role of $\Psi_{\mathcal{Y}}$. Finally, Variant5 combines both $\Psi_{\mathcal{Y}}$ and $\Psi_{\mathcal{G}}$ and significantly outperforms Variants1 and 3, which shows that domain-level information is beneficial when properly integrated with class-level classification.

6 Conclusion

This work opens a new research direction in deepfake detection by addressing a previously overlooked imbalanced learning problem that arises during detector training. Specifically, we introduce the first bi-level balanced (BB) loss that jointly tackles class-level performance imbalance and domain-level classification imbalance. In addition to extensive experimental results demonstrating consistent improvements over baseline methods, we provide rigorous theoretical support by establishing both Bayes consistency and $\mathcal{H}$ consistency. Importantly, the proposed loss can be readily integrated into a wide range of existing deepfake detection frameworks to enhance their reliability.

Limitation. A limitation of the proposed approach is that the final learning objective involves several hyperparameters that require tuning during training. **Future work**. First, beyond loss design, we plan to integrate other imbalanced learning strategies with our loss to further improve balanced performance. Second, our current study focuses on multi-domain fake data, and extending it to multi-domain real and fake data is a future direction. We also plan to generalize the method to multi-class settings (e.g., deepfake attribution) beyond binary classification. Third, beyond deepfake detection, we plan to apply the proposed loss to other applications, such as medical image analysis, where classes may correspond to normal and abnormal cases and domains may represent different levels of abnormality.

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 through testing, and we reviewed all AI-assisted manuscript changes for consistency and accuracy. We take responsibility for the final content of this work, including all texts and code produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

The details needed to reproduce our experiments are provided in the appendix, including the training setup, search space, and hyperparameters specific to each experiment described in the main paper. All source code necessary for training and evaluating our proposed BB Loss, as well as applying it to new detectors, is provided in the supplemental material via an anonymized codebase. The appendix also provides details on the model backbones, datasets, class/domain ratios, and other experimental parameters used in our evaluation.

REFERENCES

Pranjal Awasthi, Natalie Frank, Anqi Mao, Mehryar Mohri, and Yutao Zhong. Calibration and consistency of adversarial surrogate losses. *Advances in Neural Information Processing Systems*, 34:9804–9815, 2021.

Pranjal Awasthi, Anqi Mao, Mehryar Mohri, and Yutao Zhong. H-consistency bounds for surrogate loss minimizers. In *International Conference on Machine Learning*, pp. 1117–1174. PMLR, 2022.

Jonathon Byrd and Zachary Lipton. What is the effect of importance weighting in deep learning? In *International conference on machine learning*, pp. 872–881. PMLR, 2019.

Kaidi Cao, Colin Wei, Adrien Gaidon, Nikos Arechiga, and Tengyu Ma. Learning imbalanced datasets with label-distribution-aware margin loss, 2019. URL https://arxiv.org/abs/1906.07413.

Victor Chen, Madhuhaas Gottimukkala, and Jiheng Zhang. Addressing class imbalance in deepfake detection through resnet-50 ensemble with specialist models and threshold optimization. *Technical Report*, 2025.

François Chollet. Xception: Deep learning with depthwise separable convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1251–1258, 2017.

Corinna Cortes, Mehryar Mohri, and Yutao Zhong. Improved balanced classification with theoretically grounded loss functions. *NeurIPS*, 2025.

Yin Cui, Menglin Jia, Tsung-Yi Lin, Yang Song, and Serge Belongie. Class-balanced loss based on effective number of samples, 2019. URL https://arxiv.org/abs/1901.05555.

DeepFakes. Deepfakes. https://github.com/deepfakes/faceswap, 2017.

Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34:8780–8794, 2021.

Brian Dolhansky, Joanna Bitton, Ben Pflaum, Jikuo Lu, Russ Howes, Menglin Wang, and Cristian Canton Ferrer. The deepfake detection challenge (dfdc) dataset, 2020. URL https://arxiv.org/abs/2006.07397.

Jintong Gao, He Zhao, Dan dan Guo, and Hongyuan Zha. Distribution alignment optimization through neural collapse for long-tailed classification. In *Forty-first International Conference on Machine Learning*, 2024.

Google. Nano banana 2. https://gemini.google/overview/image-generation/, 2026. Accessed: 2026-08-12.

Shu Hu, Lipeng Ke, Xin Wang, and Siwei Lyu. Tkml-ap: Adversarial attacks to top-k multi-label learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 7649–7657, 2021.

Shu Hu, Xin Wang, and Siwei Lyu. Rank-based decomposable losses in machine learning: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(11):13599–13620, 2023.

Christian Internò, Robert Geirhos, Markus Olhofer, Sunny Liu, Barbara Hammer, and David Klindt. Ai-generated video detection via perceptual straightening. *NeurIPS*, 2025.

Yan Ju, Shu Hu, Shan Jia, George H Chen, and Siwei Lyu. Improving fairness in deepfake detection. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 4655–4665, 2024.

Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of stylegan. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 8110–8119, 2020.

Marek Kowalski. Faceswap. In *https://github.com/MarekKowalski/FaceSwap/*, 2018.

Yamini Sri Krubha, Aryana Hou, Braden Vester, Web Walker, Xin Wang, Li Lin, and Shu Hu. Robust ai-generated face detection with imbalanced data. *MIPR*, 2025.

Daniel Levy, Yair Carmon, John C. Duchi, and Aaron Sidford. Large-scale methods for distributionally robust optimization, 2020. URL https://arxiv.org/abs/2010.05893.

Lingzhi Li, Jianmin Bao, Hao Yang, Dong Chen, and Fang Wen. Faceshifter: Towards high fidelity and occlusion aware face swapping. *arXiv preprint arXiv:1912.13457*, pp. 2, 5, 2019.

Yuezun Li, Xin Yang, Pu Sun, Honggang Qi, and Siwei Lyu. Celeb-df: A large-scale challenging dataset for deepfake forensics, 2020. URL https://arxiv.org/abs/1909.12962.

Li Lin, Neeraj Gupta, Yue Zhang, Hainan Ren, Chun-Hao Liu, Feng Ding, Xin Wang, Xin Li, Luisa Verdoliva, and Shu Hu. Detecting multimedia generated by large ai models: A survey. *arXiv preprint arXiv:2402.00045*, 2024a.

Li Lin, Xinan He, Yan Ju, Xin Wang, Feng Ding, and Shu Hu. Preserving fairness generalization in deepfake detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 16815–16825, 2024b.

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 Computer Vision and Pattern Recognition Conference*, pp. 3503–3515, 2025.

Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection, 2018. URL https://arxiv.org/abs/1708.02002.

Honggu Liu, Xiaodan Li, Wenbo Zhou, Yuefeng Chen, Yuan He, Hui Xue, Weiming Zhang, and Nenghai Yu. Spatial-phase shallow learning: rethinking face forgery detection in frequency domain. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 772–781, 2021.

Yuchen Luo, Yong Zhang, Junchi Yan, and Wei Liu. Generalizing face forgery detection with high-frequency features. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 16317–16326, 2021.

Anqi Mao, Mehryar Mohri, and Yutao Zhong. Cross-entropy loss functions: Theoretical analysis and applications. In *International conference on Machine learning*, pp. 23803–23828. pmlr, 2023.

Aditya Menon, Harikrishna Narasimhan, Shivani Agarwal, and Sanjay Chawla. On the statistical consistency of algorithms for binary classification under class imbalance. In *International Conference on Machine Learning*, pp. 603–611. PMLR, 2013.

Aditya Krishna Menon, Sadeep Jayasumana, Ankit Singh Rawat, Himanshu Jain, Andreas Veit, and Sanjiv Kumar. Long-tail learning via logit adjustment. *ICLR*, 2021.

David Mildenberger, Paul Hager, Daniel Rueckert, and Martin J Menten. A tale of two classes: Adapting supervised contrastive learning to binary imbalanced datasets, 2025. URL https://arxiv.org/abs/2503.17024.

Huy H Nguyen, Junichi Yamagishi, and Isao Echizen. Capsule-forensics: Using capsule networks to detect forged images and videos. In *ICASSP 2019-2019 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pp. 2307–2311. IEEE, 2019.

Yunsheng Ni, Depu Meng, Changqian Yu, Chengbin Quan, Dongchun Ren, and Youjian Zhao. Core: Consistent representation learning for face forgery detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12–21, 2022.

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*, pp. 24480–24489, 2023.

Dana Pessach and Erez Shmueli. A review on fairness in machine learning. *ACM Computing Surveys (CSUR)*, 55(3):1–44, 2022.

Wenbo Pu, Jing Hu, Xin Wang, Yuezun Li, Shu Hu, Bin Zhu, Rui Song, Qi Song, Xi Wu, and Siwei Lyu. Learning a deep dual-level network for robust deepfake detection. *Pattern Recognition*, 130: 108832, 2022.

Yuyang Qian, Guojun Yin, Lu Sheng, Zixuan Chen, and Jing Shao. Thinking in frequency: Face forgery detection by mining frequency-aware clues. In *European conference on computer vision*, pp. 86–103. Springer, 2020.

Xingye Qiao and Yufeng Liu. Adaptive weighted learning for unbalanced multicategory classification. *Biometrics*, 65(1):159–168, 2009.

Jiawei Ren, Cunjun Yu, Shunan Sheng, Xiao Ma, Haiyu Zhao, Shuai Yi, and Hongsheng Li. Balanced meta-softmax for long-tailed visual recognition, 2020. URL https://arxiv.org/abs/2007.10740.

G. Research. Contributing data to deepfake detection research. Online, 2019. Accessed: 2024-04-12.

R Tyrrell Rockafellar, Stanislav Uryasev, et al. Optimization of conditional value-at-risk. *Journal of risk*, 2:21–42, 2000.

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*, pp. 1–11, 2019.

Shiori Sagawa, Pang Wei Koh, Tatsunori B. Hashimoto, and Percy Liang. Distributionally robust neural networks for group shifts: On the importance of regularization for worst-case generalization, 2020. URL https://arxiv.org/abs/1911.08731.

Ingo Steinwart. How to compare different loss functions and their risks. *Constructive Approximation*, 26(2):225–287, 2007.

Jingru Tan, Changbao Wang, Buyu Li, Quanquan Li, Wanli Ouyang, Changqing Yin, and Junjie Yan. Equalization loss for long-tailed object recognition, 2020. URL https://arxiv.org/abs/2003.05176.

Mingxing Tan and Quoc Le. Efficientnet: Rethinking model scaling for convolutional neural networks. In *International conference on machine learning*, pp. 6105–6114. PMLR, 2019.

Justus Thies, Michael Zollhofer, Marc Stamminger, Christian Theobalt, and Matthias Nießner. Face2face: Real-time face capture and reenactment of rgb videos. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2387–2395, 2016.

Justus Thies, Michael Zollhöfer, and Matthias Nießner. Deferred neural rendering: Image synthesis using neural textures. *Acm Transactions on Graphics (TOG)*, 38(4):1–12, 2019.

Arash Vahdat and Jan Kautz. Nvae: A deep hierarchical variational autoencoder. *Advances in neural information processing systems*, 33:19667–19679, 2020.

Byron C Wallace, Kevin Small, Carla E Brodley, and Thomas A Trikalinos. Class imbalance, redux. In *2011 IEEE 11th international conference on data mining*, pp. 754–763. Ieee, 2011.

Mika Westerlund. The emergence of deepfake technology: A review. *Technology innovation management review*, 9(11), 2019.

Zhiyuan Yan, Yong Zhang, Yanbo Fan, and Baoyuan Wu. Ucf: Uncovering common features for generalizable deepfake detection. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 22412–22423, 2023.

Tianbao Yang and Yiming Ying. Auc maximization in the era of big data and ai: A survey. *ACM computing surveys*, 55(8):1–37, 2022.

Zhiyong Yang, Qianqian Xu, Zitai Wang, Sicong Li, Boyu Han, Shilong Bao, Xiaochun Cao, and Qingming Huang. Harnessing hierarchical label distribution variations in test agnostic long-tail recognition. In *Proceedings of the 41st International Conference on Machine Learning*, pp. 56624–56664, 2024.

Yifan Zhang, Bingyi Kang, Bryan Hooi, Shuicheng Yan, and Jiashi Feng. Deep long-tailed learning: A survey. *IEEE transactions on pattern analysis and machine intelligence*, 45(9):10795–10816, 2023.

Zhisheng Zhong, Jiequan Cui, Shu Liu, and Jiaya Jia. Improving calibration for long-tailed recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 16489–16498, 2021.

Xiaoling Zhou and Ou Wu. Which samples should be learned first: Easy or hard? *IEEE Transactions on Neural Networks and Learning Systems*, 35(11):15264–15278, 2023.

Beier Zhu, Kaihua Tang, Qianru Sun, and Hanwang Zhang. Generalized logit adjustment: Calibrating fine-tuned models by removing label bias in foundation models. *Advances in Neural Information Processing Systems*, 36:64663–64680, 2023.

Muzhi Zhu, Chengxiang Fan, Hao Chen, Yang Liu, Weian Mao, Xiaogang Xu, and Chunhua Shen. Generative active learning for long-tailed instance segmentation. In *International Conference on Machine Learning*, pp. 62349–62368. PMLR, 2024.

A An Illustrative Figure

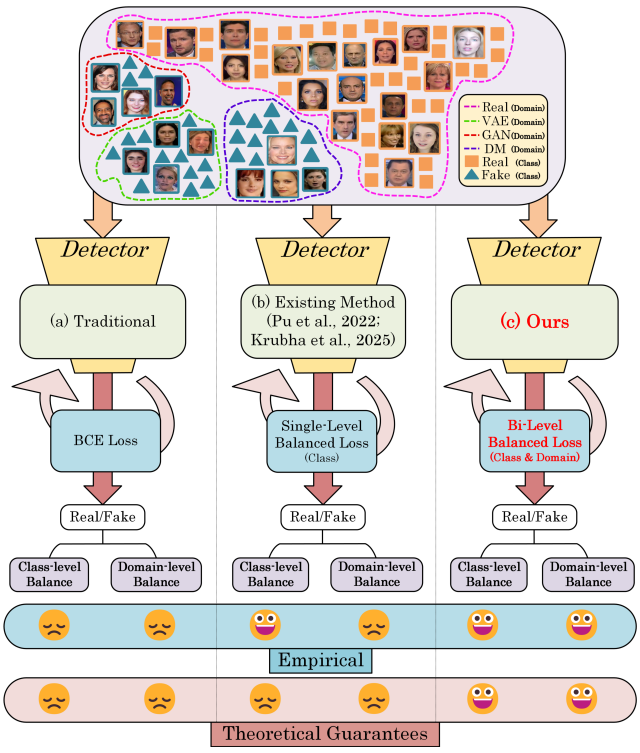

Figure 5: Comparison between our method and existing deepfake detection baselines, where fake samples are generated from VAE-, GAN-, and diffusion-based domains. (a) Conventional methods trained with BCE loss *fail* to achieve balanced performance. (b) Existing imbalance-aware baselines achieve balance *only* at the class level in practice. (c) Our method achieves **both** class-level and domain-level balance empirically **with** theoretical guarantees.

B Related Work

Deepfake Detection. The majority of existing deepfake detection approaches have followed a data-driven paradigm, relying on DNNs trained on paired authentic and manipulated content to learn discriminative artifacts indicative of forgery, as systematically summarized in recent survey Lin et al. (2024a). Most of these methods Qian et al. (2020); Liu et al. (2021); Luo et al. (2021); Yan et al. (2023); Ojha et al. (2023); Ju et al. (2024); Lin et al. (2024b); Internò et al. (2025) primarily concentrate on detector architecture design and feature representation analysis between real and fake samples, aiming to enhance overall detection accuracy and cross-dataset generalization, with binary cross-entropy (BCE) commonly adopted as the default training objective. Although such approaches have demonstrated good detection performance under controlled scenarios, they largely overlook detection bias induced by imbalanced training data distributions. As shown in a recent study Lin et al. (2025), severe imbalance between real and fake classes can lead to notable performance degradation, where detectors could tend to achieve high accuracy on one class while performing poorly on the other. *This asymmetric behavior substantially limits their reliability in practical deployment scenarios.*

Imbalanced Learning in Deepfake Detection. Only a limited number of studies have explicitly examined imbalanced learning in the context of deepfake detection Pu et al. (2022); Krubha et al. (2025); Chen et al. (2025). In particular, Pu et al. (2022); Krubha et al. (2025) integrate the binary cross-entropy loss with existing AUC-based objectives Yang & Ying (2022) to alleviate the influence of class-imbalanced training data on overall detection performance. Chen et al. (2025) suggests that ensemble strategies and a collection of training heuristics may mitigate class imbalance. However, *all of them focus exclusively on balancing real and fake classes while* ***overlooking*** *imbalance at the domain level and the impact from sample hardness*.

Imbalanced Learning in Machine Learning. Learning with imbalanced data has long been a central topic in machine learning, with a wide range of solutions proposed Zhang et al. (2023). Representative directions include data-level strategies such as re-sampling Wallace et al. (2011), data generation Qiao & Liu (2009), and data augmentation Zhu et al. (2024); model-level techniques including representation learning Gao et al. (2024), ensemble learning Yang et al. (2024), and decoupled training Zhong et al. (2021); as well as post-hoc correction methods Zhu et al. (2023). Among these, the most general and widely applicable direction lies in the design or modification of loss functions Cortes et al. (2025). Our approach follows this line of work and aims to provide the **first** *theoretically grounded and empirically effective solution* for balanced deepfake detection. More importantly, we address a **new** bi-level imbalanced learning problem. This setting is *fundamentally different* from fairness learning Pessach & Shmueli (2022). Specifically, there are **three** key distinctions: (1) our problem does not involve any protected attributes. (2) The domain variable can directly determine the class label, rather than being independent of the prediction target as assumed in many fairness learning problems. (3) Most importantly, each domain in our setting corresponds to a single class, which fundamentally differs from fairness learning scenarios where each group typically contains samples from multiple classes.

C PROOFS

We prove all theorems stated in the main text.

C.1 PROOF OF THEOREM 1

Proof. Given the relationship between $\mathcal{Y}$ and $\mathcal{G}$, both issues can be addressed through the following case analysis:

- *Case 1*: If $\hat{g}(x) = \text{real}$ and $g \in \{\text{fake}_{d_1}, ..., \text{fake}_{d_l}\}$, then $\hat{y}(x)$ must be real and y must be fake. Consequently, $\frac{\mathbf{1}_{\hat{y}(x)\neq y}}{\mathbf{1}_{\hat{g}(x)\neq g}} = 1$, implying that h^o shares the same objective as h.
- *Case 2*: If $\hat{g}(x) \in \{\text{fake}_{d_1}, ..., \text{fake}_{d_l}\}$ and $g = \text{real}$, then $\hat{y}(x)$ must be fake and y must be real. Consequently, $\frac{\mathbf{1}_{\hat{y}(x)\neq y}}{\mathbf{1}_{\hat{g}(x)\neq g}} = 1$, indicating that h^o also shares the same objective as h.
- *Case 3*: If $\hat{g}(x) \in \{\text{fake}_{d_1}, ..., \text{fake}_{d_l}\}$ and $g \in \{\text{fake}_{d_1}, ..., \text{fake}_{d_l}\}$, then $\hat{y}(x)$ must be fake and y must also be fake. As a result, $\mathbf{1}_{\hat{y}(x)\neq y} = 0$ and the term ② becomes zero, regardless of whether $\mathbf{1}_{\hat{g}(x)\neq g}$ is zero. Consequently, $\frac{\mathbf{1}_{\hat{y}(x)\neq y}}{\mathbf{1}_{\hat{g}(x)\neq g}} = 0$, implying that the score function h^o does not influence the loss.
- *Case 4*: If $\hat{g}(x) = \text{real}$ and $g = \text{real}$, then $\hat{y}(x)$ must be real and y must also be real. Consequently, $\mathbf{1}_{\hat{y}(x)\neq y} = 0$ and ② is zero, which implies $\frac{\mathbf{1}_{\hat{y}(x)\neq y}}{\mathbf{1}_{\hat{g}(x)\neq g}} = 0$. Therefore, the score function h^o does not affect the loss.

Based on the above analysis, Eq. (5) becomes

$$\Psi_{\mathcal{G}}(h, x, y, g) := \Phi^q \Bigg(\underbrace{\frac{e^{h(x,y(g),g)+\frac{\log(p(g))}{1-q}}}{\sum_{g'\in\mathcal{G}} e^{h(x,y(g'),g')+\frac{\log(p(g'))}{1-q}}}}_{③} \Bigg), \tag{9}$$

where $y(g)$ is a label–domain matching function that depends on the domain, defined as $y(g) = \text{real}$ if $g = \text{real}$ and $y(g) = \text{fake}$ if $g \in \{\text{fake}_{d_1}, ..., \text{fake}_{d_l}\}$. □

C.2 PROOF OF PROPOSITION 1

Proof. We know $\sum_{i=1}^{k} \Psi_{[i]}$ is the solution of

$$\max_{\mathbf{p}} \ \mathbf{p}^\top \overline{\Psi}, \quad \text{s.t.} \quad \mathbf{p}^\top \mathbf{1} = k, \quad \mathbf{0} \le \mathbf{p} \le \mathbf{1}.$$

We apply the Lagrangian to this problem and obtain

$$L = -\mathbf{p}^\top \overline{\Psi} - \mathbf{v}^\top \mathbf{p} + \mathbf{u}^\top (\mathbf{p} - \mathbf{1}) + \lambda(\mathbf{p}^\top \mathbf{1} - k),$$

where $\mathbf{u} \geq \mathbf{0}$, $\mathbf{v} \geq \mathbf{0}$, and $\lambda \in \mathbb{R}$ are Lagrangian multipliers. Taking the derivative with respect to $\mathbf{p}$ and setting it to zero, we have

$$\mathbf{v} = \mathbf{u} - \overline{\Psi} + \lambda \mathbf{1}.$$

Substituting this back into the Lagrangian, we obtain

$$\min_{\mathbf{u},\lambda} \ \mathbf{u}^\top \mathbf{1} + k\lambda, \quad \text{s.t.} \quad \mathbf{u} \geq \mathbf{0}, \quad \mathbf{u} + \lambda \mathbf{1} - \overline{\Psi} \geq \mathbf{0}.$$

This means

$$\sum_{i=1}^{k} \Psi_{[i]} = \min_{\lambda \in \mathbb{R}} \left\{ k\lambda + \sum_{i=1}^{m} [\Psi_i - \lambda]_+ \right\}.$$

Therefore,

$$\frac{1}{k} \sum_{i=1}^{k} \Psi_{[i]}(h, x, y, g) = \min_{\lambda \in \mathbb{R}} \left\{ \lambda + \frac{1}{k} \sum_{i=1}^{n} [\Psi(h, x_i, y_i, g_i) - \lambda]_+ \right\}.$$

□

C.3 Proof of Theorem 2

Proof. When $\alpha = 1$, the proposed surrogate loss reduces to Ψ. We will show the loss $\Psi = \Psi_{\mathcal{Y}} + \beta \Psi_{\mathcal{G}}$ is Bayes-consistent with respect to $\Psi_{BAL} := \ell^{\mathcal{Y}}_{BAL} + \beta \ell^{\mathcal{G}}_{BAL}$.

Part (i). We first prove that $\Psi_{\mathcal{Y}}$ is Bayes-consistent with respect to $\ell^{\mathcal{Y}}_{BAL}$.

Note that the predicted label is $\hat{y} = \arg\max_{y \in \mathcal{Y}} h(x, y, g)$, which depends on both x and g. Fix (x, g). Let h^*_{GLA} denote a Bayes-optimal score function minimizing the conditional GLA risk $\mathcal{R}_{GLA}(h|x, g) := \sum_{y \in \mathcal{Y}} p(y|x, g) \Psi_{\mathcal{Y}}(h(x, \cdot, g), y)$. Here $\Psi_{\mathcal{Y}}(h(x, \cdot, g), y)$ is $\Psi_{\mathcal{Y}}$ of Eq. equation 4 and its form highlights the exponential term $e^{h + \log(p(y))/(1-q)}$ in equation 4 depending on h and y. From the optimality condition of the GLA loss $\Psi_{\mathcal{Y}}$, we have that

$$\frac{e^{h^*_{GLA}(x,y,g) + \frac{\log(p(y))}{1-q}}}{\sum_{y' \in \mathcal{Y}} e^{h^*_{GLA}(x,y',g) + \frac{\log(p(y'))}{1-q}}} = \frac{(p(y|x))^{\frac{1}{1-q}}}{\sum_{y' \in \mathcal{Y}} (p(y'|x))^{\frac{1}{1-q}}}.$$

Rearranging the above equation yields

$$e^{h^*_{GLA}(x,y,g)} = \frac{\sum_{y' \in \mathcal{Y}} e^{h^*_{GLA}(x,y',g) + \frac{\log(p(y'))}{1-q}}}{\sum_{y' \in \mathcal{Y}} (p(y'|x))^{\frac{1}{1-q}}} \left(\frac{p(y|x)}{p(y)} \right)^{\frac{1}{1-q}},$$

where the factor $C(x, g) := \frac{\sum_{y' \in \mathcal{Y}} e^{h^*_{GLA}(x,y',g) + \frac{\log(p(y'))}{1-q}}}{\sum_{y' \in \mathcal{Y}} (p(y'|x))^{\frac{1}{1-q}}}$ depends on (x, g) but not on y. Hence, by Bayes' theorem, one has that $\frac{p(y|x)}{p(y)} \propto p(x|y)$ and then $e^{h^*_{GLA}(x,y,g)} \propto (p(x|y))^{\frac{1}{1-q}}$ up to a multiplicative factor independent of y. Therefore, the induced classifier satisfies

$$\hat{y}^*(x, g) = \arg\max_{y \in \mathcal{Y}} h^*_{GLA}(x, y, g) = \arg\max_{y \in \mathcal{Y}} p(x|y),$$

which is independent of g. On the other hand, from Lemma 1 Cortes et al. (2025) the Bayes classifier $h^*_{BAL}(x)$ of the balanced loss $\ell^{\mathcal{Y}}_{BAL}$ satisfies that

$$h^*_{BAL}(x) = \arg\max_{y \in \mathcal{Y}} \frac{p(y|x)}{p(y)} = \arg\max_{y \in \mathcal{Y}} p(x|y).$$

Thus, $\hat{y}^*(x, g) = h^*_{BAL}(x)$ for any $g \in \mathcal{G}$. This proves that $\Psi_{\mathcal{Y}}$ is Bayes-consistent with respect to $\ell^{\mathcal{Y}}_{BAL}$ even when the score function h and prediction rule depend on g.

Part (ii). We next show that $\Psi_{\mathcal{G}}$ is Bayes-consistent with respect to $\ell^{\mathcal{G}}_{BAL}$. From equation 9, $\Psi_{\mathcal{G}}$ is defined as

$$\Psi_{\mathcal{G}}(h, x, y, g) := \Phi^q \left(\frac{e^{h(x,y(g),g) + \frac{\log(p(g))}{1-q}}}{\sum_{g' \in \mathcal{G}} e^{h(x,y(g'),g') + \frac{\log(p(g'))}{1-q}}} \right).$$

Here $y = y(g)$ denotes a label-domain matching function. Defining $h'(x, g) = h(x, y(g), g)$, the loss $\Psi_{\mathcal{G}}$ depends only on (x, g). Applying Theorem 3 of Cortes et al. (2025), we conclude that $\Psi_{\mathcal{G}}$ is Bayes-consistent with respect to $\ell^{\mathcal{G}}_{BAL}$.

Combining Parts (i) and (ii), we conclude that $\Psi = \Psi_{\mathcal{Y}} + \beta\Psi_{\mathcal{G}}$ is Bayes-consistent with respect to $\Psi_{BAL} = \ell^{\mathcal{Y}}_{BAL} + \beta\ell^{\mathcal{G}}_{BAL}$.

□

C.4 A Counterexample

Example 1. *Consider a simplified setting in which the domain variable is ignored. The conditional risk is*

$$C_{\lambda,\alpha}(h, x) = \sum_{y\in\mathcal{Y}} p(y|x)\Big(\alpha\lambda + [\Psi_{\mathcal{Y}}(h, x, y) - \lambda]_+\Big).$$

with $\Phi^q(t) = 2(1 - \sqrt{t})$ *when* $q = 1/2$. *When the domain variable is ignored, Eq. equation 4 reduces to a label-dependent loss* $\Psi_{\mathcal{Y}}(h, x, y)$, *which is used in* $C_{\lambda,\alpha}(h, x)$. *Let* $\mathcal{Y} = \{\pm 1\}$ *with* $p(-1|x) = 0.6$ *and* $p(1|x) = 0.4$, *and class priors* $p(-1) = 0.9$ *and* $p(1) = 0.1$. *The balanced-loss Bayes rule is* $\arg\max_{y\in\{\pm1\}} \frac{p(y|x)}{p(y)}$, *which yields the balanced Bayes label* $\hat{y}^*_{BAL}(x) = 1$.

Fix $\lambda = 1.9 > 0$ *and any* $\alpha \in (0, 1)$. *Since Bayes analysis allows unbounded scores, we may choose* h *such that for a selected label* k, $h(x, k) \gg h(x, y), \forall \quad y \neq k$, *so that the softmax term in* $\Psi_{\mathcal{Y}}(h, x, y)$ *gives* $\frac{e^{h(x,k)+\frac{\log(p(k))}{1-q}}}{\sum_{y'\in\mathcal{Y}} e^{h(x,y')+\frac{\log(p(y'))}{1-q}}} \approx 1$ *and* $\frac{e^{h(x,y)+\frac{\log(p(y))}{1-q}}}{\sum_{y'\in\mathcal{Y}} e^{h(x,y')+\frac{\log(p(y'))}{1-q}}} \approx 0, \ y \neq k$. *If* $y = k$, *then* $\Psi_{\mathcal{Y}}(h, x, y) \approx 0 \leq \lambda$, *and the loss is approximately* $\alpha\lambda$. *If* $y \neq k$, *the loss is approximately* $\alpha\lambda + [\Psi_{\mathcal{Y}}(h, x, y) - \lambda]_+ \approx \alpha\lambda + 0.1$. *Therefore, the conditional risk can be approximated by* $C_{\lambda,\alpha}(h, x) \approx \alpha\lambda + 0.1\big(1 - p(k|x)\big)$. *Comparing this quantity for* $k = -1$ *and* $k = 1$, *the surrogate risk is minimized by choosing* $k = -1$, *which differs from the balanced Bayes prediction* $k = 1$.

C.5 Proof of Theorem 3

Proof. From Theorem 4 of Cortes et al. (2025), we have the following

$$\mathcal{R}_{\ell^{\mathcal{Y}}_{BAL}}(h) - \mathcal{R}^*_{\ell^{\mathcal{Y}}_{BAL}}(\mathcal{H}) + \mathcal{M}_{\ell^{\mathcal{Y}}_{BAL}}(\mathcal{H}) \leq \Gamma_{\mathcal{Y}}(\mathcal{R}_{\Psi_{\mathcal{Y}}}(h) - \mathcal{R}^*_{\Psi_{\mathcal{Y}}}(\mathcal{H}) + \mathcal{M}_{\Psi_{\mathcal{Y}}}(\mathcal{H})), \tag{10}$$

and

$$\mathcal{R}_{\ell^{\mathcal{G}}_{BAL}}(h) - \mathcal{R}^*_{\ell^{\mathcal{G}}_{BAL}}(\mathcal{H}) + \mathcal{M}_{\ell^{\mathcal{G}}_{BAL}}(\mathcal{H}) \leq \Gamma_{\mathcal{G}}(\mathcal{R}_{\Psi_{\mathcal{G}}}(h) - \mathcal{R}^*_{\Psi_{\mathcal{G}}}(\mathcal{H}) + \mathcal{M}_{\Psi_{\mathcal{G}}}(\mathcal{H})), \tag{11}$$

where $\Gamma_{\mathcal{Y}}(t) = \frac{\sqrt{2t}}{a^{\mathcal{Y}}_q}$ and $\Gamma_{\mathcal{G}}(t) = \frac{\sqrt{2t}}{a^{\mathcal{G}}_q}$. The parameters $a^{\mathcal{Y}}_q$ and $a^{\mathcal{G}}_q$ are defined as follows:

$$a^{\mathcal{Y}}_q = \begin{cases} p^{\mathcal{Y}}_{\min}, & \text{if } q = 0, \\ (p^{\mathcal{Y}}_{\min})^{\frac{1}{1-q}}\sqrt{1-q}, & \text{if } q \in (0, 1), \end{cases} \qquad a^{\mathcal{G}}_q = \begin{cases} p^{\mathcal{G}}_{\min}, & \text{if } q = 0, \\ (p^{\mathcal{G}}_{\min})^{\frac{1}{1-q}}\sqrt{1-q}, & \text{if } q \in (0, 1). \end{cases}$$

Adding equation 10 with $\beta\times$equation 11 yields

$$\mathcal{R}_{\Psi_{BAL}}(h) - \mathcal{R}^*_{\Psi_{BAL}}(\mathcal{H}) + \mathcal{M}_{\Psi_{BAL}}(\mathcal{H}) \leq \frac{\sqrt{2t_1}}{a^{\mathcal{Y}}_q} + \beta\frac{\sqrt{2t_2}}{a^{\mathcal{G}}_q}, \tag{12}$$

where $t_1 := \mathcal{R}_{\Psi_{\mathcal{Y}}}(h) - \mathcal{R}^*_{\Psi_{\mathcal{Y}}}(\mathcal{H}) + \mathcal{M}_{\Psi_{\mathcal{Y}}}(\mathcal{H})$ and $t_2 := \mathcal{R}_{\Psi_{\mathcal{G}}}(h) - \mathcal{R}^*_{\Psi_{\mathcal{G}}}(\mathcal{H}) + \mathcal{M}_{\Psi_{\mathcal{G}}}(\mathcal{H})$. Since $0 < \beta \leq 1$, we have that

$$\frac{\sqrt{2t_1}}{a^{\mathcal{Y}}_q} + \beta\frac{\sqrt{2t_2}}{a^{\mathcal{G}}_q} \leq \frac{\sqrt{2t_1} + \sqrt{2\beta t_2}}{\min\{a^{\mathcal{Y}}_q, a^{\mathcal{G}}_q\}} := \Gamma(t_1 + \beta t_2), \tag{13}$$

where $\Gamma(t) = \frac{\sqrt{2t}}{\min\{a^{\mathcal{Y}}_q, a^{\mathcal{G}}_q\}}$. Specifically, $\Gamma(t) = \frac{\sqrt{2t}}{p_{\min}}$ for $q = 0$, and $\Gamma(t) = \frac{\sqrt{2t}}{(p_{\min})^{\frac{1}{1-q}}(1-q)^{\frac{1}{2}}}$ for $q \in (0, 1)$. Combining equation 12 and equation 13 gives

$$\mathcal{R}_{\Psi_{BAL}}(h) - \mathcal{R}^*_{\Psi_{BAL}}(\mathcal{H}) + \mathcal{M}_{\Psi_{BAL}}(\mathcal{H}) \leq \Gamma(\mathcal{R}_{\Psi}(h) - \mathcal{R}^*_{\Psi}(\mathcal{H}) + \mathcal{M}_{\Psi}(\mathcal{H})).$$

Moreover, if the approximation error satisfy that $\mathcal{A}_\Psi(\mathcal{H}) = 0$, equation 12 reduces to

$$\mathcal{R}_{\Psi_{BAL}}(h) - \mathcal{R}^*_{\Psi_{BAL}}(\mathcal{H}) \leq \Gamma(\mathcal{R}_\Psi(h) - \mathcal{R}^*_\Psi(\mathcal{H})). \tag{14}$$

For notational convenience, we introduce the Conditional Value-at-Risk (CVaR) Rockafellar et al. (2000). Given a loss function $\ell : \mathcal{H} \times \mathcal{X} \times \mathcal{Y} \times \mathcal{G} \to [0, 1]$, the CVaR at level $\alpha \in (0, 1]$ is defined as

$$\mathrm{CVaR}(\ell; \alpha) = \inf_{\lambda \in \mathbb{R}} \left\{ \lambda + \frac{1}{\alpha} \mathbb{E}_{(x,y,g)\sim\mathcal{D}}[\ell - \lambda]_+ \right\}. \tag{15}$$

Let A, B be integrable random variables. For any $\lambda \in \mathbb{R}$, the map $u \mapsto (u - \lambda)_+$ is 1-Lipschitz, i.e. $|(A - \lambda)_+ - (B - \lambda)_+| \leq |A - B|$. Taking expectations and multiplying the inequality by $1/\alpha$ yields

$$\left| \left(\lambda + \frac{1}{\alpha} \mathbb{E}[A - \lambda]_+ \right) - \left(\lambda + \frac{1}{\alpha} \mathbb{E}[B - \lambda]_+ \right) \right| \leq \frac{1}{\alpha} \mathbb{E}|A - B|.$$

Taking infima over λ on both sides gives

$$|\,\mathrm{CVaR}(A; \alpha) - \mathrm{CVaR}(B; \alpha)| \leq \frac{1}{\alpha} \mathbb{E}|A - B|. \tag{16}$$

Set $A = \Psi_{BAL}(h, Z)$ and $B = \Psi_{BAL}(h^*, Z)$, where h^* is a Bayes-optimal predictor for the balanced loss Ψ_{BAL} over $\mathcal{H}$. Substituting A and B into equation 16 and combining it with equation 14, we have that

$$\begin{aligned}
&|\,\mathrm{CVaR}(\Psi_{BAL}(h, Z); \alpha) - \mathrm{CVaR}(\Psi_{BAL}(h^*, Z); \alpha)| \\
&\leq \frac{1}{\alpha} \mathbb{E}(\Psi_{BAL}(h, Z) - \Psi_{BAL}(h^*, Z)) = \frac{1}{\alpha}(\mathcal{R}_{\Psi_{BAL}}(h) - \mathcal{R}^*_{\Psi_{BAL}}(\mathcal{H})) \leq \widetilde{\Gamma}(\mathcal{R}_\Psi(h) - \mathcal{R}^*_\Psi(\mathcal{H})),
\end{aligned}$$

where $\widetilde{\Gamma}(t) = \frac{\sqrt{2t}}{\alpha p_{\min}}$ for $q = 0$, and $\widetilde{\Gamma}(t) = \frac{\sqrt{2t}}{\alpha (p_{\min})^{\frac{1}{1-q}} (1-q)^{\frac{1}{2}}}$ for $q \in (0, 1)$. It completes the proof.

□

D Implementation Details

All experiments are implemented in PyTorch and trained on an NVIDIA RTX A6000. We train all models with a batch size of 32 for a maximum 15 epochs using the SGD optimizer with a learning rate of $\eta = 5 \times 10^{-4}$. q and β are tuned from $\{0, 0.1, 0.2, ..., 1\}$. α is tuned from $\{0.1, 0.2, ..., 1\}$.

We use q_y and q_g to denote the hyperparameters q in $\Psi_\mathcal{Y}$ and $\Psi_\mathcal{G}$, respectively. The hyperparameter settings used to reproduce our experimental results are reported in Tables 7, 8, and 9.

Table 7: Hyperparameters for Table 1

Backbone	Real:Fake	FF++	AI-Face
Xception	10:1	$q_y = 0.5, q_g = 0.4, \beta = 0.1, \alpha = 0.5$	
	100:1	$q_y = 0.3, q_g = 0.5, \beta = 0.2, \alpha = 0.9$	
EfficientNet-B4	10:1	$q_y = 0.5, q_g = 0.4, \beta = 0.1, \alpha = 0.3$	
	100:1	$q_y = 0.3, q_g = 0.5, \beta = 0.2, \alpha = 0.9$	

Table 8: Hyperparameters for Figure 4 (Left)

Real:Fake	Hyperparameters
1:100	$q_y = 0.5, q_g = 0.4, \beta = 0.9, \alpha = 0.2$
1:10	$q_y = 0.1, q_g = 0.4, \beta = 1.0, \alpha = 0.8$
1:1	$q_y = 0.1, q_g = 0.2, \beta = 0.6, \alpha = 0.9$
10:1	$q_y = 0.5, q_g = 0.4, \beta = 0.1, \alpha = 0.3$
100:1	$q_y = 0.3, q_g = 0.5, \beta = 0.2, \alpha = 0.9$

Table 9: Hyperparameters for Table 3, Figure 1, and Figure 2.

Detectors	Hyperparameters
UCF	$q_y = 0.5, q_g = 0.5, \beta = 0.5, \alpha = 0.3$
CORE	$q_y = 0.1, q_g = 0.1, \beta = 0.7, \alpha = 0.1$
Capsule	$q_y = 0.1, q_g = 0.1, \beta = 0.9, \alpha = 0.7$
SPSL	$q_y = 0.5, q_g = 0.4, \beta = 0.1, \alpha = 0.5$
SRM	$q_y = 0.5, q_g = 0.4, \beta = 0.1, \alpha = 0.3$

E More Experimental Results

More Sensitive Analysis on Imbalance Ratios. Fig. 6 (Left) reports the performance of all methods on FF++ with an Xception backbone under various curated imbalance ratios. Our method consistently outperforms all baselines in terms of ACC across all ratio settings. As the imbalance ratio increases, the overall ACC of all baseline methods degrades, whereas our approach remains comparatively stable. Fig. 6 (Right) further reports the test performance of each method trained on the original, uncurated training set, confirming the effectiveness of our approach without impact on data curation.

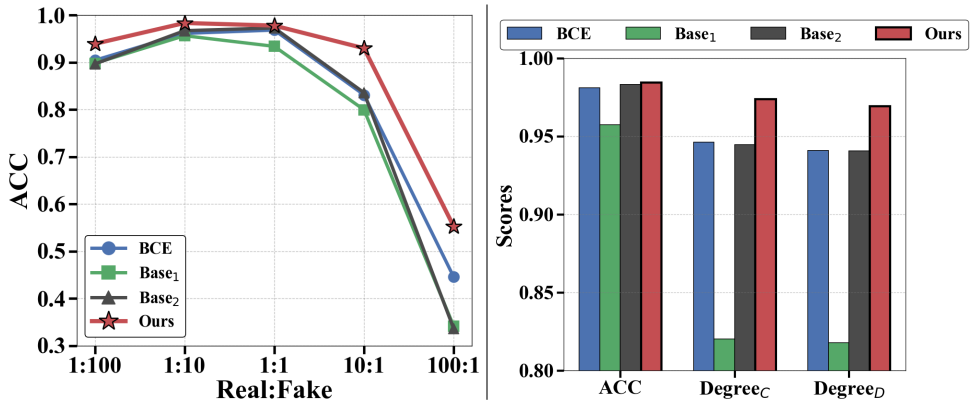

Figure 6: (**Left**) Sensitive analysis on ACC performance under curated imbalance ratios. (**Right**) The results based on the original ratio in training set.

More Plug-and-Play Experimental Results. Figure 7 shows more experimental results to demonstrate that the state-of-the-art detectors can get more balanced domain-level performance by integrating our method.

Hyperparameter Analysis. We further investigate the sensitivity of the model to the hyperparameter values q_y, q_g, β, and α in Fig. 8.

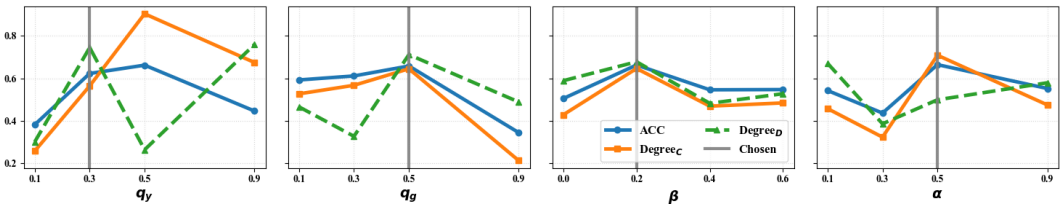

Figure 8: Sensitivity analysis of hyperparameters q_y, q_g, β, and α.

Robustness under Sample Size Reduction. When the dataset is reduced from 17218:17218 to 8609:8609, all metrics drop, which shows sensitivity to sample size. Nevertheless, in Table 10, our method consistently achieves the best performance across both settings.

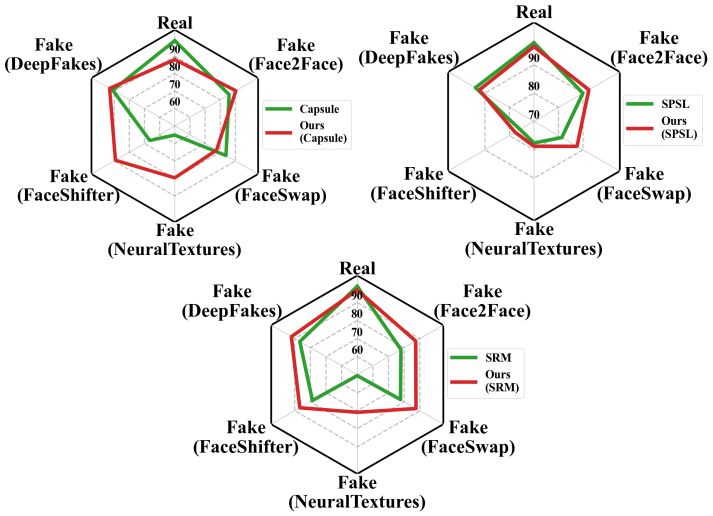

Figure 7: The flexibility and effectiveness of our approach to existing detectors.

Table 10: Performance comparison under different balanced dataset sizes.

Ratio	17218:17218				8609:8609			
Metrics	ACC	AUC	Degree$_C$	Degree$_D$	ACC	AUC	Degree$_C$	Degree$_D$
BCE	0.9684	0.9921	0.9142	0.9050	0.9619	0.9902	0.9425	0.9271
Base$_1$	0.9339	0.9770	0.9463	0.9052	0.9461	0.9958	0.9511	0.8710
Base$_2$	0.9729	0.9940	0.8949	0.8885	0.9655	0.9923	0.9557	0.9405
Ours	**0.9779**	**0.9963**	**0.9702**	**0.9603**	**0.9657**	0.9895	**0.9612**	**0.9437**

Model's ACC Under Common Corruptions. To examine how our approach affects the robustness of models, we apply a series of typical image corruptions to the testing set Lin et al. (2025), which include RC, RT, BC, HSV, GB, and JC operations. As shown in Fig. 9, our approach shows a similar behavior to the baselines without producing any new negative impacts on the robustness of models.

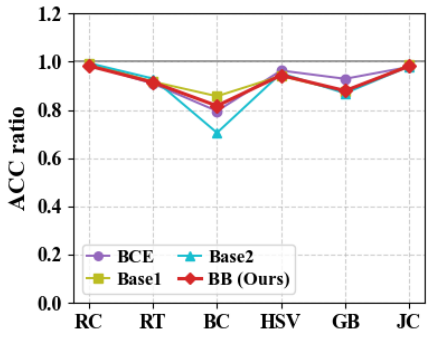

Figure 9: Model's ACC Under Common Corruptions.

Sensitivity to Decision Threshold. Since neither FF++ nor AI-Face provides an official validation split, we followed the same evaluation protocol for all methods. To address potential concerns about threshold sensitivity, we additionally evaluated BCE, GLA, and BB across multiple decision thresholds (0.1, 0.3, 0.5, 0.7, 0.9). As shown in Table 11, BB consistently achieves the best ACC, Degree$_C$, and Degree$_D$ across all thresholds, demonstrating that its gains are not simply a consequence of using the default threshold of 0.5.

Table 11: Performance comparison across different decision thresholds on FF++ (100:1).

	BCE			GLA			**BB (Ours)**		
Threshold	ACC	Degree_C	Degree_D	ACC	Degree_C	Degree_D	ACC	Degree_C	Degree_D
0.1	0.5300	0.4421	0.1973	0.8292	0.0000	0.0000	**0.6460**	**0.6166**	**0.4286**
0.3	0.4764	0.3743	0.1464	0.8256	0.1089	0.1068	**0.5867**	**0.5241**	**0.3265**
0.5	0.4458	0.3360	0.1172	0.3258	0.2286	0.0766	**0.5526**	**0.4754**	**0.2754**
0.7	0.4191	0.3030	0.0965	0.1794	0.0111	0.0025	**0.5225**	**0.4354**	**0.2391**
0.9	0.3803	0.2554	0.0682	0.1713	0.0006	0.0000	**0.4775**	**0.3753**	**0.1831**

Additional Comparison with General Imbalanced Learning Baselines. Table 12 extends the comparison in the main paper by reporting results across both 10:1 and 100:1 imbalance ratios, including additional baselines: GLA Cortes et al. (2025), Focal Lin et al. (2018), TTC Mildenberger et al. (2025), and Balanced Softmax Ren et al. (2020). TTC adapts supervised contrastive learning to binary imbalanced settings, while Balanced Softmax extends logit adjustment to long-tailed visual recognition. Our method consistently outperforms all baselines across both ratios and all metrics, demonstrating that our bi-level approach provides advantages beyond what single-level class-oriented methods can achieve.

Table 12: Comparison with general imbalanced learning baselines on FF++ across 10:1 and 100:1 ratios.

	10:1				100:1			
Method	ACC↑	AUC	Degree_C↑	Degree_D↑	ACC↑	AUC	Degree_C↑	Degree_D↑
BCE	0.8312	0.9819	0.8033	0.6620	0.4456	0.8835	0.3358	0.1172
Base_1	0.7996	0.9766	0.7658	0.6162	0.3413	0.8019	0.2128	0.0413
Base_2	0.8358	0.9785	0.8112	0.5972	0.3361	0.8697	0.2005	0.0315
GLA	0.8331	0.9346	0.8458	0.6619	0.4339	0.7210	0.3524	0.1757
Focal	0.7970	0.9793	0.7622	0.5673	0.3808	0.8482	0.2580	0.0866
TTC Mildenberger et al. (2025)	0.8336	0.9742	0.8104	0.6537	0.2534	0.7885	0.0999	0.0112
Balanced Softmax	0.8623	0.9741	0.8676	0.6950	0.4837	0.8846	0.3802	0.2175
Ours	**0.9299**	0.9781	**0.9860**	**0.8958**	**0.5526**	0.8653	**0.4754**	**0.2754**

Furthermore, Table 13 offers a thorough comparison with Balanced Softmax over several trials, further validating the consistency and supremacy of our method. In the 10:1 scenario, BB surpasses Balanced Softmax on all fronts, especially in terms of Degree_D score which is 17.49 points higher (88.11 vs. 70.62). In the tougher 100:1 regime, BB achieves a higher ACC and class-domain balance even when Balanced Softmax has a higher AUC score.

Table 13: Detailed comparison between Balanced Softmax and Ours with standard deviations.

Method	10:1				100:1			
	ACC	AUC	Degree_C	Degree_D	ACC	AUC	Degree_C	Degree_D
Balanced Softmax	86.83±0.75	97.44±0.08	87.00±0.58	70.62±2.01	44.46±3.88	86.47±2.47	33.25±4.75	17.51±5.25
Ours	**92.51±0.33**	**97.52±0.20**	**97.48±0.89**	**88.11±2.56**	**50.71±3.21**	84.01±1.82	**41.41±4.33**	**22.36±4.47**

Performance on Original FF++ Distribution. Results for all methods trained and tested on the original distribution of the FF++ dataset are shown in Table 14. In all cases, our method outperforms the competing methods. This proves the effectiveness of the proposed method even when considering the natural data distribution.

Table 14: Performance comparison on the original FF++ distribution across different backbones.

Backbone	Method	ACC↑	AUC↑	Degree$_C$↑	Degree$_D$↑
Xception	BCE	0.9812	0.9969	0.9463	0.9410
	Base$_1$	0.9575	0.9891	0.8203	0.8178
	Base$_2$	0.9831	0.9899	0.9447	0.9408
	Ours	**0.9844**	**0.9977**	**0.9738**	**0.9695**
EfficientNet-B4	BCE	0.9799	0.9965	0.9468	0.9389
	Base$_1$	0.9761	0.9960	0.9447	0.9353
	Base$_2$	0.9738	0.9952	0.8718	0.8693
	Ours	**0.9830**	**0.9972**	**0.9834**	**0.9737**