A Sanity Check for Multi-In-Domain Face Forgery Detection in the Real World

Supplementary Material

1. Details for Training and Evaluation

1.1. Training

As we mentioned, all results in this paper are reproduced based on the official code in DeepfakeBench [38]. Specifically, the original data videos are sampled to 8 frames each for training and testing. The faces in each frame are detected and cropped via Dlib [14], and 10% padding is maintained for each face image. During training, we introduce multiple data augmentations following the configuration of Effort [41], including HF (horizontal flip), BC (brightness–contrast adjustment), HSV (hue–saturation–value shift), IC (image compression), GN (Gaussian noise), MB (motion blur), CS (channel shuffle), CO (Cutout), RG (random gamma), and GB (glass blur). These augmentations are applied with preset probabilities to increase appearance diversity and improve the model's robustness to illumination changes, noise, blur, compression artifacts, and partial occlusions.

1.2. Evaluation

For a binary classifier producing a continuous prediction score $s_i \in \mathbb{R}$, let

$$\mathcal{P} = \{(s_i, y_i) \mid y_i = 1\}, \qquad \mathcal{N} = \{(s_j, y_j) \mid y_j = 0\}$$

denote the sets of positive ("fake") and negative ("real") samples.

AUC. The AUC measures the probability that a randomly chosen positive sample is assigned a higher score than a randomly chosen negative sample:

$$\mathrm{AUC} = \frac{1}{|\mathcal{P}| \cdot |\mathcal{N}|} \sum_{(s_i,1)\in\mathcal{P}} \sum_{(s_j,0)\in\mathcal{N}} \mathbb{I}(s_i > s_j) + \frac{1}{2}\mathbb{I}(s_i = s_j),$$

where $\mathbb{I}(\cdot)$ is the indicator function.

Given a decision threshold τ, the predicted label is

$$\hat{y}_i = \begin{cases} 1, & s_i \geq \tau, \\ 0, & s_i < \tau. \end{cases}$$

F-ACC. The fake accuracy is defined as the fraction of positive (fake) samples correctly classified:

$$\mathrm{Acc}_{\text{fake}} = \frac{1}{|\mathcal{N}|} \sum_{(s_j,0)\in\mathcal{N}} \mathbb{I}(\hat{y}_j = 1).$$

R-ACC. Similarly, the real accuracy measures the proportion of negative (real) samples correctly classified:

$$\mathrm{Acc}_{\text{real}} = \frac{1}{|\mathcal{P}|} \sum_{(s_i,1)\in\mathcal{P}} \mathbb{I}(\hat{y}_i = 0).$$

Algorithm 1: Deverloper for Detector (DevDet)

Input: Dataset: $S_m = \{\mathbf{X}_{real}, \mathbf{X}_{fake}\}$; Designed Detector $f(\cdot, \theta_p)$; Developer Generator $G(\cdot, \theta_g)$.

Initialize $f(\cdot)$ pretrained on S_m;
Initialize dataset for Optimizing Developer
$S_1 = \{S_{HF}, S_{ER}\}$
training stage 1 for developer
for $\mathbf{x} \sim S_1$ **do**
  predict developer δ_{dev} based on $\mathbf{x}$
  $\delta_{\text{dev}} = G(\mathbf{x}, \theta_g)$
  apply developer to image
  $\tilde{\mathbf{x}} = \mathbf{x} + \epsilon\delta_{\text{dev}}$
  predict real/fake
  $y_p = f(\tilde{\mathbf{x}}, \theta_p)$
  compute developer loss
  $L_{\text{dev}} = -\left(\hat{y}\log(y_p) + (1-\hat{y})\log(1-y_p)\right).$
  compute overall loss of stage 1
  $L_{o1} = L_{\text{dev}} + \lambda_{tv} L_{\text{tv}}$
  update θ_g based on L_{o1} via backpropagation
prepare DoseDict $\mathbf{D}$
$\min_{\mathbf{D},\{\boldsymbol{\alpha}_i\}} \sum_{i=1}^{N} \left(\frac{1}{2}\|\mathbf{z}_i - \mathbf{D}\boldsymbol{\alpha}_i\|_2^2 + \lambda\|\boldsymbol{\alpha}_i\|_1\right)$
Dose Adaptive Fine-Tuning (DAFT) for $f(\cdot, \theta_p)$
for $\mathbf{x} \sim S_m$ **do**
  adaptively compute dose
  $\epsilon_a = \mathrm{Norm}(1 - ||\mathbf{z} - \mathbf{D}^*\boldsymbol{\alpha}^*(\mathbf{z})||_2)$
  apply adaptive developer to image
  $\tilde{\mathbf{x}} = \mathbf{x} + \epsilon_a\delta_{\text{dev}}$
  predict real/fake
  $y_p = f(\tilde{\mathbf{x}}, \theta_p)$
  compute DAFT loss
  $L_{\text{daft}} = -\left(\hat{y}\log(y_p) + (1-\hat{y})\log(1-y_p)\right).$
  update θ_p based on L_{o1} via backpropagation

Output: Trained $G(\cdot, \theta_g)$, $f(\cdot)$, and $\mathbf{D}$.

Based on these definitions, it could be clearly observed that AUC represents the relative division between real and fake samples. For example, supposing all fake samples are detected as 0.9 while all real samples are detected as 0.8, the AUC will be **100%**. However, for real-world detection with a fixed accuracy threshold (maybe $\tau = 0.5$), the above case will have an accuracy of 0.5, which is equal to random guess. Therefore, AUC cannot accurately measure the real-world application performance of deepfake detector, especially in MID-FFD scenario, where domain distinction surpasses the real-fake distinction, making the absolute real-fake decision even challenger.

Table 5. Comparison across different sample selection strategies.

Methods	Volume	Datasets				Avg
		FF++	CDF	DFDCP	WDF	
Base	-	0.7724	0.7696	0.7840	0.7233	0.7623
HF-only	Small	0.7803	0.7762	0.7917	0.7351	0.7708
	Large	0.8123	0.7975	0.8103	0.7372	0.7893
HF+HR	Small	0.7831	0.7715	0.7931	0.7386	0.7716
	Large	0.8144	0.8205	0.8153	0.7702	0.8051.
All	Small	0.8181	0.8042	0.7980	0.7453	0.7914
	Large	0.8495	0.8457	0.8593	0.8078	0.8405
HF+ER (Ours)	Small	0.8921	0.8535	0.8717	0.8530	0.8675
	Large	0.8950	0.8785	0.8745	0.8755	0.8809

2. Algorithm

The Algorithm of the DevDe is shown in Alg. 1. In the algorithm, we concisely illustrate the two stages of the proposed DevDet during training, including FFDev optimization, DoseDict fitting, and DAFT for the pretrained detector.

3. Further Experiments

3.1. Effect of Sample Selection Strategy for Optimizing FFDev

In this paper, we select Hard-Fake (HF) and Easy-Real (ER) to optimize the Face Forgery Developer. To demonstrate the effect of maintaining real while enhancing fake, we design the following ablation variants: 1) HF-only: Investigating the effectiveness of maintaining real. 2) HF and HR (Hard Real): Attempting to enhance both real and fake at the same time. 3) All: using unspecified training samples to retrain the FFDev. All variants are considered with two versions, that is, the small set (5000 samples) and the large set (20000 samples). Subsequent to these variants, the DAFT is conducted in the same way as usual. The experiments are conducted based on Effnb4 and protocol 1. As shown in Tab. 5, the first observation is that the results show limited sensitivity to the volume of training samples for the HF-only, HF+HR, and HF+ER models. This suggests that a small number of challenging samples are sufficient to effectively represent the forgery trace for training. Then, it can be observed that HF-only and HF+HR exhibit marginal improvement due to no Real sample as a relative reference for preservation. All performs better while still being inferior to HF+ER.

3.2. Samples Volume for Training DoseDict

As a sensitive hyperparameter, the number of samples to fit a DoseDict via dictionary learning is crucial for the accuracy and generalization ability of the predicted dose. Here, we apply a range of sample numbers as an ablation study, which is shown in Fig. 6. It can be observed that, in the early stages of data augmentation, both generalization and detection accuracy show a certain degree of improvement. However, as the dataset grows too large, generalization performance gradually saturates, and the accuracy of MID experiences a slight decline. Therefore, this study selects 6000 as the optimal volume.

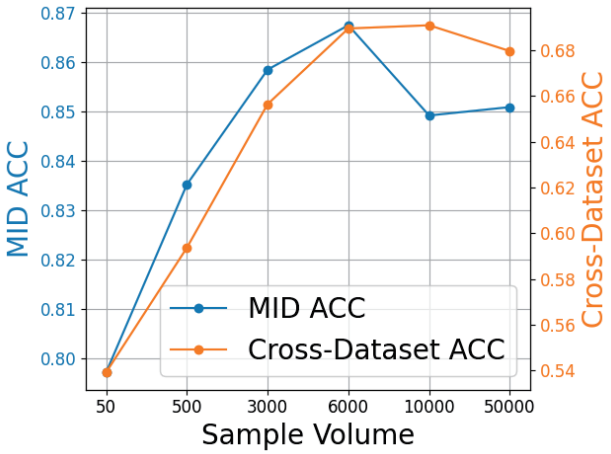

Figure 6. Effect of sample volume.

Table 6. Assigned dose $\epsilon_a \in [0, 1.5]$ under different evaluation scenarios.

ϵ_a	In-Train	In-Test	DF40	DiffFace
Mean	0.16	0.37	0.19	0.07
Std	0.27	0.49	0.29	0.05

3.3. Influence of ACC Threshold

As threshold sensitivity can fundamentally affect the final accuracy, especially in MID scenario, as shown in Fig. 7, the baseline methods exhibit substantial performance variations across different decision thresholds, which may introduce practical challenges when deploying these methods in real-world scenarios. In contrast, DevDet consistently maintains superior performance and demonstrates more stable behavior under different threshold settings. These results suggest that DevDet is less sensitive to threshold selection and therefore provides more reliable detection performance.

3.4. Side-by-side comparison between In-Domain (ID), Cross-Domain (CD), and the proposed Multiple In-Domain (MID)

The main evaluation difference between MID and CD is: MID focuses on ACC with large-scale diverse data, rather than limited-in-domain (ID) or OOD (CD) AUC with limited training data. We provide **side-by-side paradigm comparison in Tab. 7**.

Table 7. Comparison of different evaluation paradigms, including MID, In-Domain (ID), and Cross-Dataset (CD). F_n denotes distinct datasets. The motivation for introducing MID-FFD beyond ID and CD is discussed in Sec. 3 and illustrated in Fig. 2 of the main paper.

Paradigm	Train Set	Test Set	D-Num.	Metric	Protocol	Motivation
ID	F_1	F_1	1	AUC (rel.)	One-for-One	Fitting One
CD	F_1	F_2, F_3	≥ 2	AUC (rel.)	One-for-Other	Generalize Other
MID	$F_1, F_2, \ldots, F_n$	$F_1, F_2, \ldots, F_n$	$\geq n$	ACC (abs.)	Many-for-Many	Fitting Multiple

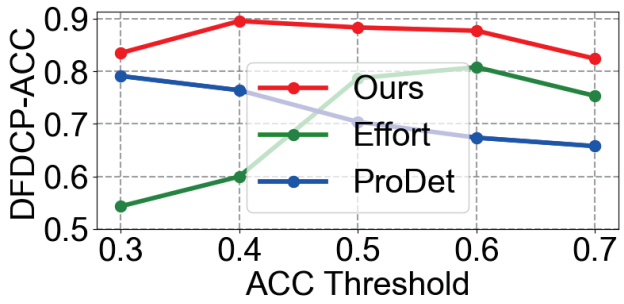

Figure 7. Performance with Multiple ACC Threshold (%)

3.5. Assign Dose (ϵ_a) on Unseen Data

With the continuous expansion of training datasets, defining a strictly "unseen" scenario becomes increasingly challenging. To better analyze the behavior of the assigned dose ϵ_a under different generalization settings, we design a series of experiments that gradually transition from seen to unseen distributions, including: in-domain training and testing, unseen fake samples (DF40), and fully unseen fake *and* real samples (DiffFace). As reported in Tab. 6, the assigned dose ϵ_a becomes more active when evaluating in-domain test data and unseen fake samples, indicating that the mechanism effectively responds to distribution shifts within related domains. In contrast, ϵ_a approaches zero on DiffFace, where both fake and real samples are entirely unseen during training. This observation reveals a potential failure case of our method: when the training data volume is insufficient to adequately cover the target distribution, FFDev may become redundant for completely unseen inputs. In such scenarios, the mechanism is rarely activated, suggesting that further improvements in data coverage or robustness may be necessary to fully exploit the benefits of FFDev under extreme domain shifts.

4. Discussion with Concurrent Study

Interestingly, we observe a concurrent study [27] on AIGC detection that shares almost ***identical motivation and observations*** with this work. First, this coincidence strongly corroborates the importance and generality of the MID problem formulated in this paper. Second, due to the intrinsic variability of human faces and the comparatively subtle nature of forgery artifacts, domain discrepancies in face forgery are more likely to undermine the absolute decision boundary between real and fake samples. Consequently, we observe that deepfake detection already exhibits substantial boundary collapse under relatively limited multi-domain conditions, whereas AIGC detection typically requires a much larger scale of domain diversity for the phenomenon to manifest clearly. In summary, the findings of both this work and the concurrent study [27] highlight the critical role of MID detection in both AIGC and deepfake scenarios, while simultaneously revealing their differences in granularity and domain sensitivity.