LET DETECTORS TALK: DETECTOR-GUIDED DATA SELECTION FOR DEEPFAKE DETECTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Deepfake detectors often struggle to generalize to unseen forgery patterns. We investigate whether adapting the use of available training data to a detector's evolving weaknesses can improve generalization. To this end, we alternate detector training with the selection of samples for subsequent training stages. The difficulty is deciding what to select: although labeled samples reveal where the current detector struggles, similar prediction scores may conceal different failure patterns and offer limited guidance on how to address them. We use the samples the detector finds difficult as anchors for identifying its current weaknesses. To characterize these anchors beyond the detector's own scores, we select auxiliary detectors based on their detection performance and response redundancy. Their responses provide multiple views of both the anchors and the samples available in the training pool. We then select additional training samples that exhibit similar response patterns to the anchors, retrain the target detector, and repeat the process with updated anchors. Experiments on six cross-domain test sets demonstrate the effectiveness of our method and show competitive performance.

1 INTRODUCTION

Deepfake detection research has produced a wide range of detectors (Qian et al., 2020; Lin et al., 2024), datasets (Rossler et al., 2019; Li et al., 2020b), and benchmarks (Yan et al., 2023b; 2024b). Much of the current research focuses on how to train an effective detector given a dataset (Kashiani et al., 2025; Cui et al., 2025; Sun et al., 2026). Yet a detector trained on existing data may still make errors on new data (Li et al., 2025; Yermakov et al., 2026). This raises a natural question that how can we improve an existing detector when it makes such errors?

A straightforward way to improve the detector is to add labeled samples to its training set and retrain it (Furuhashi et al., 2024). The challenge is deciding which samples to add, since their usefulness depends on what the detector has yet to learn. A sample that improves one detector may not be helpful to another, and its value may change as the detector is updated (Lei et al., 2026; Chou et al., 2026). The current detector's prediction errors reveal cases it struggles with, which we use as hard anchors. A score-based baseline then retrieves same-label candidates according to their prediction-score distance from each anchor. However, as shown in Fig. 1(a), two same-label candidates may be similarly close in score to the same anchor, making it difficult to prioritize one using score distance alone.

To compare candidates with hard anchors beyond the current detector's score, we seek additional views of the same images. To this end, we examine how detectors trained on different datasets and built on different backbones respond to a shared set of images. We observe that detectors trained on the same dataset tend to show similar response patterns, whereas patterns vary across detectors trained on different datasets. This variation suggests that detector responses can offer different views of the detection cues in these images when comparing candidates with hard anchors.

In this paper, we seek reliable and complementary views for candidate retrieval. We therefore select a fixed group of expert detectors with high AUCs on the shared reference set and low pairwise response similarity. We represent each sample by the differences between each expert's response and that of the current detector, then retrieve candidates whose response-difference vectors are close to those of the hard anchors. As illustrated in Fig. 1(b), candidates at similar score distances from

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.

a hard anchor can have different distances in response-difference space, allowing us to prioritize which candidates to add.

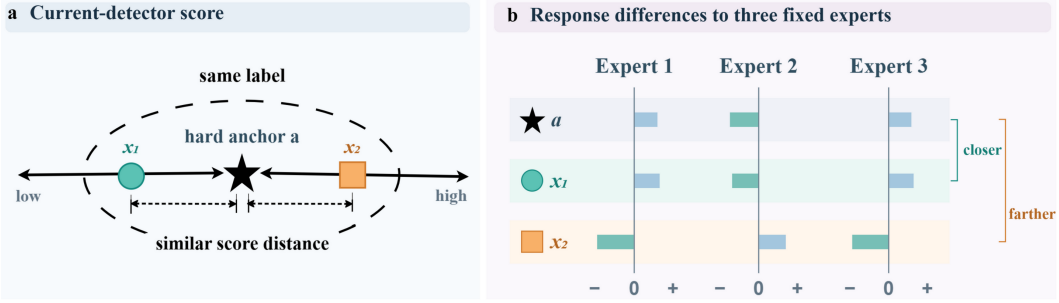

Figure 1: Conceptual illustration of candidate retrieval. (a) Two same-label candidates have similar current-detector score distances from a hard anchor. (b) Their response-difference vectors place x_1 closer to the anchor than x_2.

We add the selected samples to the detector's training set and retrain it. Before the next selection round, we re-identify hard anchors and recompute response-difference vectors using the updated detector. With the same target backbone, anchor rule, and added-sample budget, response-difference retrieval yields a higher mean AUC across six external test sets than current-detector score retrieval.

Our main contributions are as follows:

- We provide a perspective on data selection in the detector retraining stage by treating identification of the current detector's hard cases and comparison between labeled candidates and them as separate steps.
- We provide a novel detection evidence space construction method by utilizing expert detectors. We use this space to capture relationships between samples and guide the selection of additional training data.
- Experiments show that, under the same target backbone, anchor rule, and addition budget, our method achieves a higher mean AUC across six external test sets than current-detector score retrieval.

2 Related Work

2.1 Deepfake Datasets and Benchmarks

Deepfake detection research has produced many datasets and benchmarks for training and evaluating detectors. FaceForensics++ (Rossler et al., 2019), Celeb-DF (Li et al., 2020b), and DFDC (Dolhansky et al., 2020) provide face-forgery data for training and evaluation. DF40 (Yan et al., 2024b), HydraFake (Tan et al., 2026), and ScaleDF (Wang et al., 2025) broaden the range of forgery methods and data sources. DeepfakeBench (Yan et al., 2023b) and ForensicHub (Du et al., 2025). provide benchmarking frameworks for deepfake detection. Together, these resources provide diverse data for training and evaluating deepfake detectors.

2.2 Generalizable Deepfake Detection

Deepfake detectors draw on different backbones and forensic cues. Xception and EfficientNet provide classification backbones (Chollet, 2017; Tan & Le, 2019). F3-Net and SRM examine frequency features (Qian et al., 2020; Luo et al., 2021); warping-artifact detectors exploit resolution inconsistencies (Li & Lyu, 2019); and Face X-Ray and SBI use blending traces or self-blended training examples (Li et al., 2020a; Shiohara & Yamasaki, 2022). Identity-based approaches examine inconsistencies in facial identity (Dong et al., 2022; Huang et al., 2023) or address identity leakage (Dong et al., 2023). UCF disentangles common forgery features from method-specific patterns (Yan et al., 2023a). DCL and SeeABLE use contrastive learning (Sun et al., 2022; Larue et al., 2023); LSDA

and CDFA expand forgery variation through latent-space and image-space augmentation, respectively (Yan et al., 2024a; Lin et al., 2024); and D^3 learns from discrepancy signals (Yang et al., 2025). For broader AI-generated image detection, DDA aligns pixel- and frequency-level data to reduce bias (Chen et al., 2025), while Effort decomposes the feature space to preserve pretrained knowledge (Yan et al., 2025). ForensicsAdapter adapts pretrained features for face forgery detection (Cui et al., 2025), while ForensicHub trains different backbones under a shared protocol (Du et al., 2025). These detectors vary in training data and backbones. We select high-performing auxiliary detectors with less redundant responses and use their scores to characterize these anchors beyond the detector's own score

2.3 Data Selection for Improving Existing Detectors

Curriculum learning schedules training examples by difficulty (Song et al., 2024); tutor-student learning adjusts their loss weights using visual features and training history (Lei et al., 2026). DEEPDISTAL selects a representative subset of an existing deepfake dataset using active learning (Rana et al., 2024). For continued detector training, Park et al. (2026) maintain difficulty-aware caches of incoming examples, and Furuhashi et al. (2024) select new images using the current detector's confidence. Our work separates identifying the target detector's hard cases from selecting candidates to address them. We use its errors as anchors. Auxiliary detectors provide additional views of both the anchors and labeled candidates beyond the target detector's own scores, allowing us to compare their responses.

3 Methodology

The overall pipeline of our proposed approach is illustrated in Fig. 3. We first select a group of expert detectors and keep them fixed. In each round, the current detector identifies hard cases on a labeled diagnosis set. We use response differences between the current detector and the experts to compare these cases with labeled candidates from the training pool. We then add the selected candidates to the training set and retrain the detector. The updated detector guides the next round of selection.

3.1 Expert Detector Selection

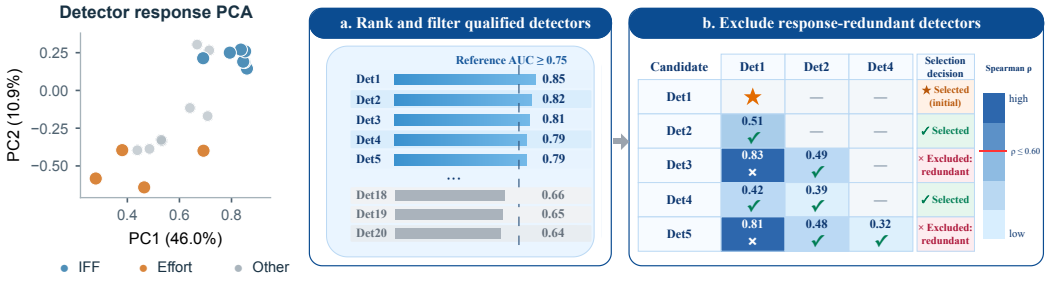

Figure 2: Detector responses and expert selection. Left: PCA of 20 detectors' responses to the same images (blue: IFF; orange: Effort; gray: others). Middle (a): detectors ranked and filtered by reference AUC, with a threshold of 0.75. Right (b): a qualified detector is retained only if its Spearman rank correlation with every selected expert is at most 0.60. AUCs are rounded and omitted detectors are marked by an ellipsis; PCA is used for visualization only.

To choose experts that provide different views for comparing samples, we examine how 20 pretrained detectors score the same images. The ForensicHub detectors follow the IFF training protocol, using the same training datasets but differing in backbone or training strategy (Du et al., 2025). The Effort variants use the same backbone and training method but were trained on different datasets (Yan et al., 2025). In the left panel of Fig. 2, we use principal component analysis (PCA)

to project each detector's scores across these images into two dimensions; nearby points suggest similar response patterns. The IFF detectors lie relatively close together, while the Effort variants are more spread out. Together, these patterns suggest that detectors trained on the same datasets may respond similarly, while those trained on different datasets may provide distinct responses. We therefore select well-performing detectors whose responses differ from one another.

As illustrated in panels (a) and (b) of Fig. 2, we first look for detectors with competitive detection performance, measured by AUC. To compare their responses, we compute the Spearman rank correlation between each pair of detectors on the same images, lower correlation indicates less similar response patterns. In our experiments, we retain detectors with AUC at least 0.75 and consider them in descending AUC order. A detector is added to the expert group only if its correlation with every selected expert is at most 0.60. This yields ConvNextSmall, Effort_FF, and DDA (Det1, Det2, and Det4 in the figures). The full selection results are reported in Appendix Table 8.

3.2 Hard Anchors and Candidate Retrieval

As shown in Step 1 of Fig. 3, the current detector identifies hard anchors in a labeled diagnosis set. We choose the decision threshold that maximizes accuracy on this set. Within each class, we rank misclassified images by their distance to the threshold and retain the closest ones, up to 10% of the diagnosis images of that class.

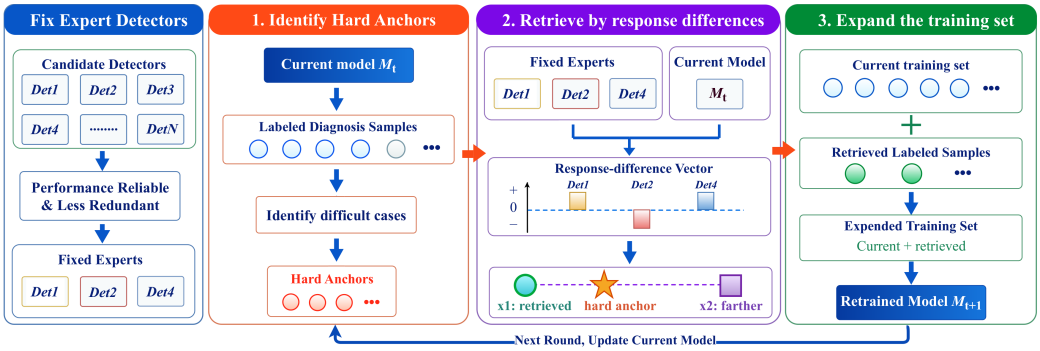

Figure 3: Data selection pipeline. A fixed expert group complements the current detector: misclassified diagnosis images become hard anchors (Step 1), expert–current response guide same-label candidate retrieval (Step 2), and the retrieved images expand the training set before retraining (Step 3). The updated detector starts the next round while the experts remain fixed.

Step 2 of Fig. 3 shows how the fixed experts help compare hard anchors with candidate images. Before iterative selection, we construct each expert's rank mapping from its scores on the labeled training pool. This maps the expert's response to $[0, 1]$ according to its rank among these scores. Both the experts and their rank mappings remain fixed across rounds.

For an image x at round t, let $r_j(x)$ denote the rank-calibrated response of expert j and $p_t(x)$ the current detector's fake-class probability. We standardize both scores on the labeled diagnosis set $\mathcal{A}$ before computing their difference:

$$g_{t,j}(x) = \frac{r_j(x) - \mu_{\mathcal{A}}(r_j)}{\sigma_{\mathcal{A}}(r_j)} - \frac{p_t(x) - \mu_{\mathcal{A}}(p_t)}{\sigma_{\mathcal{A}}(p_t)}. \tag{1}$$

Here, $\mu_{\mathcal{A}}$ and $\sigma_{\mathcal{A}}$ denote the mean and standard deviation on $\mathcal{A}$. Each difference describes an expert's response relative to that of the current detector. We standardize these differences once more on $\mathcal{A}$ and combine one value per expert into an image's response-difference vector. Within a round, the same normalization statistics are applied to hard anchors and candidates. After retraining, the current detector's predictions and the corresponding normalization statistics are recomputed, while the expert responses remain fixed. The exact normalization formula and implementation details are given in Appendix A.3. For each unused labeled candidate, we find its nearest hard anchor with the same label in response-difference space. We use the Euclidean distance to this anchor to rank

candidates within each class, with smaller distances taking priority. The class quota is filled from this ranked list across all anchors, rather than assigning a separate quota to each anchor. Each candidate is counted only once, even if it is close to several anchors. In the lower part of Step 2 in Fig. 3, x_1 is closer to the hard anchor than x_2, so it is prioritized for selection. This distance measures similarity in detector responses.

3.3 Iterative Data Selection and Model Retraining

Given a fixed group of auxiliary detectors that provide additional views of the same images, we alternate between sample selection and target-detector training. In each round, the current detector identifies misclassified images in the labeled diagnosis set as hard anchors. We compare these anchors with the remaining labeled candidates using the auxiliary detectors' responses relative to the current detector. Candidates are ranked by their response-difference distance to the nearest anchor with the same label.

Following Step 3 of Fig. 3, we add equal numbers of the highest-ranked real and fake candidates to the training set and remove them from the candidate pool. We reinitialize the classifier and train it on the original images together with all samples selected so far, while keeping the image backbone fixed. The retrained classifier becomes the current detector for the next round, we repeat these steps, iteratively updating the detector and the samples selected for training.

4 Experiments

4.1 Experimental Setup

Datasets. The labeled training pool combines ScaleDFTrain (Wang et al., 2025) and HydraTrain (Tan et al., 2026). We initially train the target detector on 29,356 class-balanced images from this pool; the remaining images are available as candidates for addition. We draw a labeled diagnosis set A and a separate held-out set B from ScaleDFVal, HydraJpgVal, and Celeb-DF v2 (Li et al., 2020b). Set A provides hard anchors. We also evaluate on six cross-domain sets (Tan et al., 2026): DeepFaceLab (Liu et al., 2023), InfiniteY (Jiang et al., 2025), Dreamina (Dreamina AI, 2025), HailuoAI (Hailuo AI, 2025), GPT-4o (OpenAI, 2025), and FFIW (Zhou et al., 2021). Set B and the six cross-domain sets are used only for evaluation. Additional dataset details are provided in Appendix A.4.

Target detector and experts. Our default target detector uses a pretrained DINOv3 ViT-7B/16 backbone (Siméoni et al., 2025) with a trainable classifier on its frozen image features. The expert detectors selected in Section 3.1 remain fixed throughout data selection.

Selection and retraining. In each matched comparison, selection methods start from the same initial detector. We add 5,000 labeled images per round for 50 rounds, with equal numbers from each class, and train a new classifier on the cumulative training set after each addition. Further training details appear in Appendix A.4.

Existing detectors. As performance references, we evaluate Xception (Chollet, 2017), SPSL (Liu et al., 2021a), SegFormer-B3 and its average-pooling variant (Xie et al., 2021), SwinSmall (Liu et al., 2021b), ConvNextSmall (Liu et al., 2022), CDFA (Lin et al., 2024), ForensicsAdapter (Cui et al., 2025), Effort_FF (Yan et al., 2025), and DDA (Chen et al., 2025). The IFF checkpoints are from ForensicHub (Du et al., 2025). Their backbones, training data, and checkpoint sources are listed in Appendix Table 7.

Comparisons and metrics. Table 1 reports the initial detector before any samples are added, followed by four data selection variants. Each variant is named by its anchor rule and candidate-matching criterion. "Rand. / score" samples anchors by class from the diagnosis set; "Conf. err. / score" and "Bound. err. / score" use misclassified images farthest from and closest to the decision threshold, respectively. These three variants rank same-label candidates by current-detector score distance to an anchor. "Bound. err. / diff." uses the boundary-error anchor rule but ranks candidates by response-difference vector distance. The four variants start from the same initial detector and use the same added-sample budget. We report AUC for diagnosis set A, held-out set B, and each of the six external test sets. The "Average" column is the mean across the six external sets.

4.2 Overall Data Selection Performance

Performance across evaluation sets. Starting from the initial detector's external average AUC of 85.0%, selecting random anchors and matching candidates by the current detector's score raises it to 86.6% after adding training images (Table 1). Using misclassified images as anchors improves AUC on diagnosis set A and held-out set B relative to random anchors, but the two score-based variants reach only 86.6% and 86.9% on the external average. With the same boundary-error anchor rule, response-difference retrieval raises the external average to 90.7%, compared with 86.9% for score-based retrieval. This suggests that finding the detector's errors helps, but how candidates are matched to those errors matters more for performance on the external sets.

Table 1: Performance of existing detectors and the DINOv3 ViT-7B/16 target with different data selection methods (AUC). In the lower block, "Initial model" shows the target before data selection; the remaining rows show its results after each method adds the same number of images to the training set. Results for the target are mean ± standard deviation across three initializations. Gray highlights our method.

	A	B	External test sets						
Method	Diagnosis set	Held-out set	DeepFace Lab	InfiniteY	Dreamina	HailuoAI	GPT-4o	FFIW	Average
Existing detectors									
Xception	73.11	72.87	68.11	87.64	79.13	70.64	70.38	64.91	73.47
SPSL	72.23	72.44	68.89	89.87	81.67	73.06	70.70	65.40	74.93
SegFormer-B3	70.65	70.92	74.17	97.94	72.16	70.07	72.63	73.74	76.79
SegFormer-B3 (avg.)	73.78	73.96	76.84	96.16	75.00	68.86	72.25	74.42	77.25
SwinSmall	72.86	73.54	73.26	99.23	86.77	80.76	77.32	75.19	82.09
ConvNextSmall	77.53	77.88	75.45	99.13	88.36	85.80	82.38	74.30	84.24
CDFA	74.77	74.14	79.71	74.71	83.59	63.52	34.11	74.32	68.33
ForensicsAdapter	76.35	76.96	90.31	59.95	50.20	54.70	49.51	79.46	64.02
Effort_FF	82.93	82.96	93.33	64.24	49.46	62.92	41.23	82.72	65.65
DDA	80.63	79.75	67.49	98.45	97.19	97.54	89.27	59.02	84.83
Data selection methods									
Initial model	91.9 ± 0.4	92.0 ± 0.2	76.8 ± 3.6	86.3 ± 4.1	98.5 ± 0.4	96.2 ± 1.2	78.9 ± 2.0	73.5 ± 2.7	85.0 ± 1.5
Rand. / score	92.2 ± 0.1	92.2 ± 0.2	78.1 ± 0.7	88.9 ± 3.9	99.0 ± 0.3	97.7 ± 0.6	82.0 ± 0.6	73.7 ± 1.3	86.6 ± 1.0
Conf. err. / score	93.7 ± 0.4	93.5 ± 0.6	77.4 ± 2.3	93.1 ± 0.9	99.1 ± 0.2	97.2 ± 1.0	79.6 ± 2.0	73.0 ± 0.9	86.6 ± 0.8
Bound. err. / score	94.2 ± 0.2	93.9 ± 0.2	76.8 ± 1.1	94.5 ± 1.6	99.0 ± 0.1	98.5 ± 0.5	78.8 ± 1.0	73.9 ± 0.4	86.9 ± 0.5
Bound. err. / diff.	94.6 ± 0.1	94.5 ± 0.2	83.5 ± 0.8	97.6 ± 0.2	99.1 ± 0.1	99.0 ± 0.0	86.8 ± 1.5	78.4 ± 0.8	90.7 ± 0.5

Performance as training images are added. Figure 4 tracks external average AUC throughout iterative selection. From the first measured checkpoint at 25,000 added images through the final budget of 250,000, response-difference retrieval has a higher mean AUC than score-based retrieval under both confident-error and boundary-error anchor rules. The advantage therefore appears early and is observed with either anchor rule, supporting the role of candidate matching throughout the selection process.

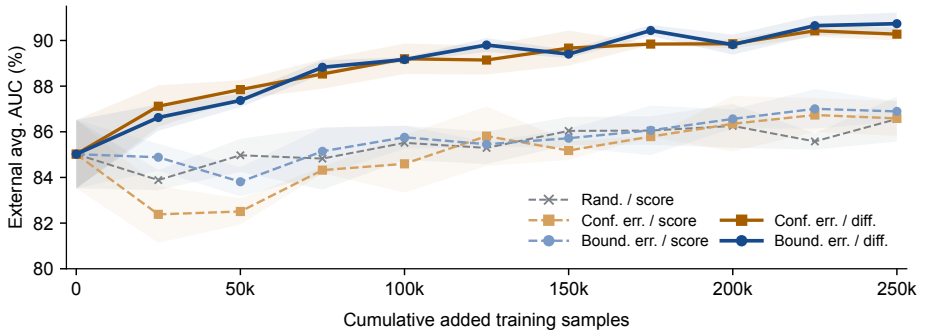

Figure 4: External average AUC as training images are added. Curves show the mean across three initializations of the DINOv3 ViT-7B/16 target; shaded bands show the standard deviation.

4.3 Ablation Study and Analysis

We first compare different expert groups, then examine how candidates are matched to anchors and which errors serve as anchors. All comparisons use the same initial target detector, sample budget, and retraining protocol. We report AUC on held-out set B and the average AUC across the six external test sets.

Expert Group Composition. We select experts with competitive detection performance and less redundant responses. The expert groups in V2-V7 all satisfy these criteria but differ in their members and sizes, while V1 uses only the current detector's score for candidate matching. Their external average AUCs range from 89.7% to 90.9%, compared with 86.9% for V1 (Table 2). The improvement is therefore observed across several qualified expert groups.

Table 2: Data selection variants with different expert groups (AUC). V1 uses no experts. V7 (gray) is the variant used in our main experiments.

ID	K	Matching / expert group	Held-out B	External avg.
V1	–	Bound. err. / score	93.9	86.9
V2	2	ConvNextSmall + Effort_FF	94.2	90.6
V3	2	Effort_FF + DDA	94.4	89.7
V4	2	ConvNextSmall + DDA	94.2	90.3
V5	3	Effort_FF + DDA + SwinSmall	94.7	90.5
V6	3	ConvNextSmall + DDA + ForensicsAdapter	94.2	90.9
V7	3	ConvNextSmall + DDA + Effort_FF	94.5	90.7

We further compare V5 and V6 with the default group V7, retaining two experts and replacing the third. V5 shares DDA and Effort_FF with V7 and replaces ConvNextSmall with SwinSmall. These two detectors have highly similar responses, with a Spearman correlation of 0.833. V6 shares ConvNextSmall and DDA with V7 and replaces Effort_FF with ForensicsAdapter; the two alternative detectors also show similar responses, with a correlation of 0.727. Both correlations exceed our redundancy threshold, so each pair is not selected together in the same expert group. V5 and V6 reach external average AUCs of 90.5% and 90.9%, respectively, compared with 90.7% for V7. These results suggest that detectors with similar responses can serve as alternatives in the tested expert groups.

Matching Candidates to Hard Anchors. With the boundary-error anchor rule fixed, matching candidates directly by the experts' scores reaches 90.3% external average AUC, compared with 86.9% using the current detector's score (Table 3). Computing the differences between expert and current-detector scores further raises it to 90.7%. Thus, expert scores are useful for matching candidates even without computing response differences, while the differences provide a smaller additional gain.

Table 3: Methods for matching candidates to hard anchors. The boundary-error anchor rule is fixed. Gray highlights our method.

Matching criterion	Held-out B	External avg.
score	93.9	86.9
Expert scores	94.2	90.3
diff.	94.5	90.7

Table 4: Rules for choosing hard anchors. Gray highlights our method.

Anchor rule	Matching	Held-out B	External avg.
Conf. err.	score	93.5	86.6
	diff.	94.2	90.3
Bound. err.	score	93.9	86.9
	diff.	94.5	90.7

Choosing Hard Anchors. We compare confident-error and boundary-error anchors under each matching method (Table 4). With score-based retrieval, choosing boundary errors raises held-out B AUC from 93.5% to 93.9% and external average AUC from 86.6% to 86.9%. With response-difference retrieval, the corresponding results rise from 94.2% to 94.5% and from 90.3% to 90.7%. Boundary-error anchors therefore give a modest improvement with either matching method.

4.4 Further Analysis

Different Initial Training Sets. Different initial training images and random seeds can produce detectors with different errors and responses, which in turn affect their hard anchors and candidate matches. For the default setting, we sample 29,356 images from the multi-source training pool and train initial detectors with different random seeds. To test a different starting forgery domain, we train another initial detector on the same number of FF++ images (Rossler et al., 2019). We keep the diagnosis and evaluation sets, the underlying training pool, and the selection and retraining procedure the same, including the number of rounds.

Table 5 shows that the two initial training sets produce detectors with different starting performance on both held-out B and the external average. After the same number of selection rounds, the gains over their respective initial models also differ. With response-difference retrieval, the FF++ model gains 13.3 percentage points on held-out B and 10.5 on the external average, compared with 2.5 and 5.7 for the multi-source model. Despite these differences in starting performance and subsequent gains, our method achieves higher final AUC than score retrieval on both measures from either starting point.

Table 5: Data selection from different initial training sets. AUC is reported as mean $\pm$ standard deviation across three initializations.

Initial training set	Anchor rule / retrieval	Held-out B	External avg.
Multi-source	Initial model	92.0 ± 0.2	85.0 ± 1.5
	Bound. err. / score	93.9 ± 0.2	86.9 ± 0.5
	Bound. err. / diff.	94.5 ± 0.2	90.7 ± 0.5
FF++	Initial model	80.9 ± 0.3	79.8 ± 0.7
	Bound. err. / score	93.2 ± 0.3	87.8 ± 0.4
	Bound. err. / diff.	94.2 ± 0.1	90.3 ± 0.3

Different Target Backbones. To examine whether the method works with different target backbones, we repeat data selection and retraining with four target detectors while keeping the expert group fixed (Table 6). The procedure remains the same; each backbone supplies its own image embeddings, detector scores, and hard anchors. Adding samples selected with random anchors and score retrieval does not always improve performance: the external average decreases for CLIP and MetaCLIP. At the final sample budget, score retrieval with boundary-error anchors performs better than score retrieval with random anchors across all four backbones on the external sets. With the same boundary-error anchors, response-difference retrieval brings further gains on both held-out B and the external sets. The gains vary across backbones, but response-difference retrieval outperforms score retrieval in each case.

Table 6: Data selection with different target backbones. The last three columns report AUC changes from the initial model in percentage points.

Backbone	Evaluation	Initial AUC	Rand. / score	Bound. / score	Bound. / diff.
DINOv3 ViT-B/16	Held-out B	85.1	$+0.1$	$+1.3$	$+2.0$
	External avg.	75.8	$+1.7$	$+2.1$	$+8.7$
DINOv3 ViT-7B/16	Held-out B	92.0	$+0.2$	$+1.9$	$+2.5$
	External avg.	85.0	$+1.6$	$+1.9$	$+5.7$
CLIP ViT-L/14	Held-out B	92.8	$+0.4$	$+0.6$	$+1.0$
	External avg.	83.0	-0.4	$+0.4$	$+3.8$
MetaCLIP ViT-B/32	Held-out B	86.9	$+0.2$	$+0.7$	$+2.6$
	External avg.	76.6	-0.6	$+0.6$	$+5.5$

5 CONCLUSION

This work investigated whether adapting sample selection to a deepfake detector's evolving weaknesses can improve detection performance. We used samples the detector found difficult as anchors, then characterized them through the responses of auxiliary detectors selected for detection performance and low response redundancy. Selecting training samples with similar response patterns allowed each subsequent training stage to focus on weaknesses that the target detector's scores alone might obscure. Experiments on six cross-domain test sets demonstrate the effectiveness of this iterative strategy and show competitive performance. These results suggest that performance can benefit from choosing available training data according to the detector's current failure patterns, rather than treating all difficult samples as equally informative.

Limitations and Future Work. Our method is constrained by the coverage of the existing training pool: it cannot select samples representing forgery patterns that are absent from that pool. At the same time, this constraint highlights its practical value, as the method makes more effective use of available training data without requiring additional forgery samples. Future work could extend the framework to dynamically expanding training pools that incorporate newly emerging forgery patterns, and examine whether adaptive sample selection remains effective when the original pool has limited coverage.

AI USE STATEMENT

We used generative AI tools to assist with manuscript drafting and editing, preparation of figures and plotting code, and discussion of experimental results. The authors take responsibility for the final text, code, figures, and reported scientific claims.

ETHICS STATEMENT

This work uses existing labeled image datasets and pretrained detectors to improve deepfake detection. False positives and false negatives can affect the people depicted in images, so detector outputs should not be treated as definitive judgments of authenticity. Use of the underlying image data should respect privacy and the original dataset access conditions.

REPRODUCIBILITY STATEMENT

The expert selection criteria and candidate retrieval procedure are described in Sections 3.1 and 3.2. Detector architectures, training data, and checkpoint sources are listed in Table 7, and the complete expert screening is reported in Table 8. Appendix A.3 specifies hard-anchor identification, rank calibration, response normalization, same-label candidate ranking.

Appendix A.4 records dataset sizes and roles, pretrained backbone checkpoints, classifier architecture, training hyperparameters, and the iterative selection schedule. It also specifies the random seeds, evaluation protocol, and aggregation of results into means and sample standard deviations. Matched selection comparisons use the same initial detector, training procedure, and added-sample budget. The held-out and external test sets are used only for evaluation and do not guide selection, checkpoint choice, or stopping. Computational costs and measurement boundaries are reported in Appendix A.5, while Appendices A.6 and A.7 provide paired comparisons and detailed statistics for the main-text results.

OVERVIEW OF APPENDIX

Appendix A.1 describes detector architectures, training data, and checkpoint sources, while Appendix A.2 documents the complete expert screening. Appendix A.3 specifies response differences and candidate retrieval. Appendix A.4 provides the experimental protocol, and Appendix A.5 summarizes computational costs. Appendices A.6 and A.7 report paired comparisons across initializations and full statistics for the compact result tables, respectively.

REFERENCES

Ruoxin Chen, Junwei Xi, Zhiyuan Yan, Ke-Yue Zhang, Shuang Wu, Jingyi Xie, Xu Chen, Lei Xu, Isabel Guan, Taiping Yao, and Shouhong Ding. Dual data alignment makes AI-generated image detector easier generalizable. *Advances in Neural Information Processing Systems*, 38, 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.

Yuxuan Chou, Tao Yu, Wen Huang, ZhangYuHeng, Tao Dai, and Shu-Tao Xia. Improving deepfake detection with reinforcement learning-based adaptive data augmentation. *Proceedings of the AAAI Conference on Artificial Intelligence*, 40(5):3381–3389, 2026. doi: 10.1609/aaai.v40i5.37334.

Xinjie Cui, Yuezun Li, Ao Luo, Jiaran Zhou, and Junyu Dong. Forensics adapter: Adapting CLIP for generalizable face forgery detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 19207–19217, 2025.

Brian Dolhansky, Joanna Bitton, Ben Pflaum, Jikuo Lu, Russ Howes, Menglin Wang, and Cristian Canton Ferrer. The DeepFake detection challenge (DFDC) dataset. *arXiv preprint arXiv:2006.07397*, 2020.

Shichao Dong, Jin Wang, Renhe Ji, Jiajun Liang, Haoqiang Fan, and Zheng Ge. Implicit identity leakage: The stumbling block to improving deepfake detection generalization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3994–4004, 2023.

Xiaoyi Dong, Jianmin Bao, Dongdong Chen, Ting Zhang, Weiming Zhang, Nenghai Yu, Dong Chen, Fang Wen, and Baining Guo. Protecting celebrities from deepfake with identity consistency transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9468–9478, 2022.

Dreamina AI. Dreamina. `https://jimeng.jianying.com`, 2025. Accessed: 2025-07-25.

Bo Du, Xuekang Zhu, Xiaochen Ma, Chenfan Qu, Kaiwen Feng, Zhe Yang, Chi-Man Pun, Jian Liu, and Ji-Zhe Zhou. ForensicHub: A unified benchmark & codebase for all-domain fake image detection and localization. In *Advances in Neural Information Processing Systems, Datasets and Benchmarks Track*, volume 38, 2025. doi: 10.52202/085713-2167.

Yoshihiko Furuhashi, Junichi Yamagishi, Xin Wang, Huy H. Nguyen, and Isao Echizen. Exploring active data selection strategies for continuous training in deepfake detection. In *Proceedings of the International Conference of the Biometrics Special Interest Group*, pp. 1–5, 2024. doi: 10.1109/BIOSIG61931.2024.10786748.

Hailuo AI. Hailuo Video. `https://hailuoai.com/video/create`, 2025. Accessed: 2025-07-25.

Baojin Huang, Zhongyuan Wang, Jifan Yang, Jiaxin Ai, Qin Zou, Qian Wang, and Dengpan Ye. Implicit identity driven deepfake face swapping detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4490–4499, 2023.

Liming Jiang, Qing Yan, Yumin Jia, Zichuan Liu, Hao Kang, and Xin Lu. InfiniteYou: Flexible photo recrafting while preserving your identity. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 10898–10907, 2025.

Hossein Kashiani, Niloufar Alipour Talemi, and Fatemeh Afghah. Freqdebias: Towards generalizable deepfake detection via consistency-driven frequency debiasing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8775–8785, 2025.

Nicolas Larue, Ngoc-Son Vu, Vitomir Struc, Peter Peer, and Vassilis Christophides. SeeABLE: Soft discrepancies and bounded contrastive learning for exposing deepfakes. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 21011–21021, 2023.

Zhanhe Lei, Zhongyuan Wang, Jikang Cheng, Baojin Huang, Yuhong Yang, Zhen Han, Chao Liang, and Dengpan Ye. Tutor-student reinforcement learning: A dynamic curriculum for robust deepfake detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 41436–41445, 2026.

Hanzhe Li, Jiaran Zhou, Yuezun Li, Baoyuan Wu, Bin Li, and Junyu Dong. Freqblender: Enhancing deepfake detection by blending frequency knowledge. *Advances in Neural Information Processing Systems*, 37:44965–44988, 2024.

Lingzhi Li, Jianmin Bao, Ting Zhang, Hao Yang, Dong Chen, Fang Wen, and Baining Guo. Face X-Ray for more general face forgery detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5001–5010, 2020a.

Yuezun Li and Siwei Lyu. Exposing DeepFake videos by detecting face warping artifacts. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pp. 46–52, 2019.

Yuezun Li, Xin Yang, Pu Sun, Honggang Qi, and Siwei Lyu. Celeb-df: A large-scale challenging dataset for deepfake forensics. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3207–3216, 2020b.

Zhaolun Li, Jichang Li, Yinqi Cai, Junye Chen, Xiaonan Luo, Guanbin Li, and Rushi Lan. Fakeradar: Probing forgery outliers to detect unknown deepfake videos. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 13382–13392, 2025.

Yuzhen Lin, Wentang Song, Bin Li, Yuezun Li, Jiangqun Ni, Han Chen, and Qiushi Li. Fake it till you make it: Curricular dynamic forgery augmentations towards general deepfake detection. In *European Conference on Computer Vision*, pp. 104–122. Springer, 2024.

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, 2021a.

Kunlin Liu, Ivan Perov, Daiheng Gao, Nikolay Chervoniy, Wenbo Zhou, and Weiming Zhang. DeepFaceLab: Integrated, flexible and extensible face-swapping framework. *Pattern Recognition*, 141: 109628, 2023. doi: 10.1016/j.patcog.2023.109628.

Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9992–10002, 2021b.

Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. A convnet for the 2020s. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 11976–11986, 2022.

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.

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.

OpenAI. Addendum to GPT-4o system card: 4o image generation. `https://openai.com/index/gpt-4o-image-generation-system-card-addendum/`, 2025.

Hayoung Park, Jinhee Park, Choongsang Cho, and Guisik Kim. Efficient domain-incremental deepfake detection via difficulty-aware curriculum learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pp. 8693–8703, 2026.

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.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139, pp. 8748–8763, 2021.

Md Shohel Rana, Mohammad Nur Nobi, and Andrew Sung. DEEPDISTAL: Deepfake dataset distillation using active learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pp. 7723–7730, 2024.

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. doi: 10.1109/ICCV.2019.00009.

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

Oriane Siméoni, Huy V. Vo, Maximilian Seitzer, Federico Baldassarre, Maxime Oquab, Cijo Jose, Vasil Khalidov, Marc Szafraniec, Seungeun Yi, Michaël Ramamonjisoa, Francisco Massa, Daniel Haziza, Luca Wehrstedt, Jianyuan Wang, Timothée Darcet, Théo Moutakanni, Leonel Sentana, Claire Roberts, Andrea Vedaldi, Jamie Tolan, John Brandt, Camille Couprie, Julien Mairal, Hervé Jégou, Patrick Labatut, and Piotr Bojanowski. DINOv3. arXiv:2508.10104, 2025.

Wentang Song, Yuzhen Lin, and Bin Li. Towards generic deepfake detection with dynamic curriculum. In *2024 IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 4500–4504, 2024. doi: 10.1109/ICASSP48485.2024.10448345.

Jiamu Sun, Zhiyuan Yan, Ke-Yue Zhang, Taiping Yao, and Shouhong Ding. DFD-HR: Generalizable deepfake detection via hierarchical routing learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13984–13995, 2026.

Ke Sun, Taiping Yao, Shen Chen, Shouhong Ding, Jilin Li, and Rongrong Ji. Dual contrastive learning for general face forgery detection. *Proceedings of the AAAI Conference on Artificial Intelligence*, 36(2):2316–2324, 2022.

Hao Tan, Jun Lan, Zichang Tan, Senyuan Shi, Aijian Liu, Chuanbiao Song, Huijia Zhu, Weiqiang Wang, Jun Wan, and Zhen Lei. Veritas: Generalizable deepfake detection via pattern-aware reasoning. In *International Conference on Learning Representations*, 2026.

Mingxing Tan and Quoc V. Le. EfficientNet: Rethinking model scaling for convolutional neural networks. In *International Conference on Machine Learning*, pp. 6105–6114, 2019.

Wenhao Wang, Longqi Cai, Taihong Xiao, Yuxiao Wang, and Ming-Hsuan Yang. Scaling laws for deepfake detection. *arXiv preprint arXiv:2510.16320*, 2025.

Enze Xie, Wenhai Wang, Zhiding Yu, Anima Anandkumar, Jose M. Alvarez, and Ping Luo. Segformer: Simple and efficient design for semantic segmentation with transformers. In *Advances in Neural Information Processing Systems*, volume 34, pp. 12077–12090, 2021.

Hu Xu, Saining Xie, Xiaoqing Ellen Tan, Po-Yao Huang, Russell Howes, Vasu Sharma, Shang-Wen Li, Gargi Ghosh, Luke Zettlemoyer, and Christoph Feichtenhofer. Demystifying CLIP data. In *International Conference on Learning Representations*, 2024.

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, 2023a.

Zhiyuan Yan, Yong Zhang, Xinhang Yuan, Siwei Lyu, and Baoyuan Wu. Deepfakebench: A comprehensive benchmark of deepfake detection. *Advances in Neural Information Processing Systems*, 36:4534–4565, 2023b.

Zhiyuan Yan, Yuhao Luo, Siwei Lyu, Qingshan Liu, and Baoyuan Wu. Transcending forgery specificity with latent space augmentation for generalizable deepfake detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8984–8994, 2024a.

Zhiyuan Yan, Taiping Yao, Shen Chen, Yandan Zhao, Xinghe Fu, Junwei Zhu, Donghao Luo, Chengjie Wang, Shouhong Ding, Yunsheng Wu, and Li Yuan. Df40: Toward next-generation deepfake detection. In *Advances in Neural Information Processing Systems*, volume 37, pp. 29387–29434, 2024b. doi: 10.52202/079017-0925.

Zhiyuan Yan, Jiangming Wang, Peng Jin, Ke-Yue Zhang, Chengchun Liu, Shen Chen, Taiping Yao, Shouhong Ding, Baoyuan Wu, and Li Yuan. Orthogonal subspace decomposition for generalizable AI-generated image detection. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267, pp. 70268–70288, 2025.

Yongqi Yang, Zhihao Qian, Ye Zhu, Olga Russakovsky, and Yu Wu. D^3: Scaling up deepfake detection by learning from discrepancy. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 23850–23859, 2025.

Andrii Yermakov, Jan Cech, Jiri Matas, and Mario Fritz. Deepfake detection that generalizes across benchmarks. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 773–783, 2026.

Tianfei Zhou, Wenguan Wang, Zhiyuan Liang, and Jianbing Shen. Face forensics in the wild. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 5778–5788, 2021.

A APPENDIX

Appendix Overview

The appendix is organized into four groups:

- **Expert Detectors** — detector provenance and expert screening.
  - Appendix A.1: Detector Details — architectures, training data, and checkpoint sources for the 20 candidate detectors.
  - Appendix A.2: Expert Detector Selection — reference AUC ranking and response-correlation screening.
- **Selection Procedure** — retrieval and experimental protocol.
  - Appendix A.3: Response Differences and Candidate Retrieval — anchor selection, response normalization, and same-label retrieval.
  - Appendix A.4: Experimental Details — data roles, training settings, and evaluation protocol.
- **Resources and Comparisons** — computational cost and paired results.
  - Appendix A.5: Computational Cost — feature extraction and experiment GPU hours.
  - Appendix A.6: Paired Selection Comparisons Across Initializations — matched-seed differences at the final sample budget.
- **Detailed Results** — additional statistics for the main-text tables.
  - Appendix A.7: Full Statistics for Compact Result Tables — matching and anchor ablations, expert groups, and initialization and backbone comparisons.

A.1 Detector Details

Table 7: Provenance of the candidate detectors.

Detector	Architecture / backbone	Detector training data	Checkpoint source
ConvNextSmall	ConvNeXt-Small	IFF-Protocol	ForensicHub family
ResNet101	ResNet-101	IFF-Protocol	ForensicHub family
ResNet101 (avg.)	ResNet-101	IFF-Protocol	ForensicHub family
SPSL (Liu et al., 2021a)	Xception (RGB+phase)	IFF-Protocol	ForensicHub family
SegFormer-B3	SegFormer-B3	IFF-Protocol	ForensicHub family
SegFormer-B3 (avg.)	SegFormer-B3	IFF-Protocol	ForensicHub family
SwinSmall	Swin-Small	IFF-Protocol	ForensicHub family
UnivFD (Ojha et al., 2023)	CLIP ViT-L/14	ProGAN (20 classes)	Official release
Xception	Xception	IFF-Protocol	ForensicHub family
CDFA (Lin et al., 2024)	SwinV2-B	FF++ + pseudo-fakes	Our implementation
DDA (Chen et al., 2025)	DINOv2 ViT-L/14	DDA-aligned MSCOCO	Official DDA release
Effort_Chameleon (Yan et al., 2025)	CLIP ViT-L/14	Chameleon (SD v1.4)	Official Effort release
Effort_FF (Yan et al., 2025)	CLIP ViT-L/14	FF++ (c23)	Official Effort release
Effort_SDV14 (Yan et al., 2025)	CLIP ViT-L/14	GenImage (SD v1.4)	Official Effort release
Effort_UFD (Yan et al., 2025)	CLIP ViT-L/14	UFD	Official implementation
F3Net (Qian et al., 2020)	Xception + FAD	FF-NT (FF++, c23)	DeepfakeBench v1.0.1 release
ForensicsAdapter (Cui et al., 2025)	CLIP-L/14 + ViT-T	FF++ (c23)	Official release
FreqBlender (Li et al., 2024)	EfficientNet-B4	FF++ real + pseudo-fakes	Official release
SBI (Shiohara & Yamasaki, 2022)	EfficientNet-B4	FF++ real + pseudo-fakes	Our implementation
SRM (Luo et al., 2021)	Dual-branch Xception	FF-F2F (FF++, c23)	DeepfakeBench v1.0.1 release

A.2 Expert Detector Selection

Table 8: Reference performance and sequential screening of the candidate detectors.

Rank	Detector	Reference AUC ↑	AUC gate	$\rho_{\max}$ ↓	Screening decision
1	**ConvNextSmall**	0.848	Pass	–	**Selected (1st)**
2	**Effort_FF**	0.821	Pass	0.510	**Selected (2nd)**
3	SwinSmall	0.808	Pass	0.833	Excluded: redundant
4	**DDA**	0.795	Pass	0.425	**Selected (3rd)**
5	SegFormer-B3 (avg.)	0.794	Pass	0.810	Excluded: redundant
6	ForensicsAdapter	0.784	Pass	0.727	Excluded: redundant
7	SegFormer-B3	0.782	Pass	0.770	Excluded: redundant
8	SPSL	0.768	Pass	0.762	Excluded: redundant
9	Xception	0.755	Pass	0.730	Excluded: redundant
10	CDFA	0.726	Fail	–	Excluded by AUC
11	ResNet101 (avg.)	0.711	Fail	–	Excluded by AUC
12	ResNet101	0.705	Fail	–	Excluded by AUC
13	Effort_UFD	0.698	Fail	–	Excluded by AUC
14	UnivFD	0.694	Fail	–	Excluded by AUC
15	Effort_SDV14	0.690	Fail	–	Excluded by AUC
16	SRM	0.677	Fail	–	Excluded by AUC
17	Effort_Chameleon	0.665	Fail	–	Excluded by AUC
18	F3Net	0.658	Fail	–	Excluded by AUC
19	FreqBlender	0.649	Fail	–	Excluded by AUC
20	SBI	0.638	Fail	–	Excluded by AUC

For the configuration reported here, detectors are scanned in descending order of reference AUC, so the highest-ranked qualified detector is the initial expert. Every candidate satisfying AUC ≥ 0.75 is evaluated and selected only when its maximum Spearman Rank correlation with all previously selected experts is at most 0.60. The complete screening yields three experts. The symbol "–" indicates that no correlation test was required because the detector was either the initial expert or failed the AUC gate; gray rows denote the resulting fixed expert detector group. Reference AUCs are computed from the detectors' raw scores and real/fake labels on the labeled training pool. Rank correlations use their rank-transformed responses on the same images.

All 20 detectors are evaluated on the same images from the labeled training pool as provided. Their raw scores are oriented so that higher values indicate stronger evidence of a fake image.

A.3 Response Differences and Candidate Retrieval

At round t, let M_t denote the current detector and $\mathcal{A}$ the labeled diagnosis set. The decision threshold maximizes accuracy on $\mathcal{A}$. From each class's misclassified images, we keep those closest to the threshold, up to 10% of the diagnosis images of that class (rounded up); if fewer errors are available, we keep all of them. Let $\mathcal{H}_t$ denote the resulting hard anchors.

For an image x, the expert responses $r_j(x) \in [0,1]$ in equation 1 are rank-calibrated, with $j \in \{1,2,4\}$ indexing the selected experts. Each expert's rank mapping is fixed using its scores on the labeled training pool. We then standardize each expert's differences $g_{t,j}$ on $\mathcal{A}$ to obtain the response-difference vector:

$$\boldsymbol{\phi}_t(x) = \left(\frac{g_{t,j}(x) - \mu_{\mathcal{A}}(g_{t,j})}{\sigma_{\mathcal{A}}(g_{t,j})} \right)_{j=1,2,4}. \tag{2}$$

A zero standard deviation is replaced by one. Within a round, the same statistics from $\mathcal{A}$ are applied to hard anchors and candidates; as M_t changes, its scores and the vectors are recomputed.

For an unused labeled candidate x from the training pool, let $y(x)$ be its real-or-fake label. Its distance to the nearest same-label anchor is

$$d_t(x) = \min_{a \in \mathcal{H}_t :\, y(a)=y(x)} \|\boldsymbol{\phi}_t(x) - \boldsymbol{\phi}_t(a)\|_2 . \tag{3}$$

Candidates are ranked by this distance within each class. Each class's quota is filled across all its anchors, without reserving a fixed number of candidates for any one anchor. Each candidate is counted once, even if it is close to multiple anchors. Distance measures similarity of response relationships; it is not a direct estimate of training value.

A.4 Experimental Details

Data roles. The labeled training pool contains 5,823,422 ScaleDFTrain images and 47,841 HydraTrain images (5,871,263 in total). The multi-source initialization samples 29,356 images from this pool, equally balanced between real and fake; these samples are removed from the pool of additional candidates. The alternative FF++ initialization also contains 29,356 images and uses FF++ images from the same underlying pool. The diagnosis set A (23,197 images) and held-out set B (23,222 images) draw from ScaleDFVal, HydraJpgVal, and CDFv2. A supplies labeled cases for anchor identification, the accuracy-maximizing decision threshold for boundary-error anchors, and response-normalization statistics. B and the six external test sets (DeepFaceLab, InfiniteY, Dreamina, HailuoAI, GPT-4o, and FFIW) are used only for evaluation. Candidate labels are available for same-label retrieval and loss calculation. The split names denote the data assets used in this experiment; the reported counts and A/B partition describe our local protocol, rather than published split sizes. Different initial sets leave different remaining candidates, although both experiments start from the same underlying labeled pool.

Training and selection. The main target detector uses frozen 4,096-dimensional DINOv3 ViT-7B/16 image features with a trainable $4,096$–512–128–2 multilayer perceptron. Each matched configuration starts from the same initial detector. After adding samples, the classifier is reinitialized and trained on the cumulative labeled set for five epochs using AdamW with batch size 256, learning rate 10^{-3}, zero weight decay, and dropout 0.1. The feature extractor and selected expert detectors remain fixed. We perform 50 rounds, adding 5,000 previously unused images in each round (2,500 per class), for 250,000 new training images. Retrieval methods select candidates using their nearest anchor of the same label, with the same per-class quotas.

Target backbone weights. The frozen image encoders use pretrained DINOv3 (Siméoni et al., 2025), CLIP (Radford et al., 2021), and MetaCLIP (Xu et al., 2024) Hugging Face checkpoints: `facebook/dinov3-vit7b16-pretrain-lvd1689m`, `facebook/dinov3-vitb16-pretrain-lvd1689m`, `openai/clip-vit-large-patch14`, and `facebook/metaclip-b32-400m`, respectively for DINOv3 ViT-7B/16, DINOv3 ViT-B/16, CLIP ViT-L/14, and MetaCLIP ViT-B/32. The two DINOv3 backbones reuse extracted features; CLIP and MetaCLIP compute frozen image features online during training.

Selection variants. Table labels follow the anchor rule / candidate-matching criterion notation defined in the main text. "Uncert." selects anchors from A with predicted fake-class probabilities closest to 0.5, whether correctly classified or not; its "score" and "diff." variants use the same anchors. High loss / loss selects anchors with the highest per-sample binary cross-entropy and matches candidates by their losses. Expert scores matches candidates using the selected experts' responses. The default configuration is Bound. err. / diff. All retrieval variants enforce the same-label candidate constraint.

Evaluation and reporting. We report area under the ROC curve (AUC), which does not depend on the threshold used to define boundary errors. A and B are evaluated each round, whereas the six external sets are evaluated for the initial detector and every five rounds, including the last. External average AUC is the equal-weight mean of six test-set AUCs, not an AUC on pooled images. We cross initial-set seeds 42–44 with classifier-training seeds 42–43. For each initialization we first average the two training-seed results, then report the mean and sample standard deviation across three initializations. The held-out and external results do not guide selection, checkpoint choice, or stopping.

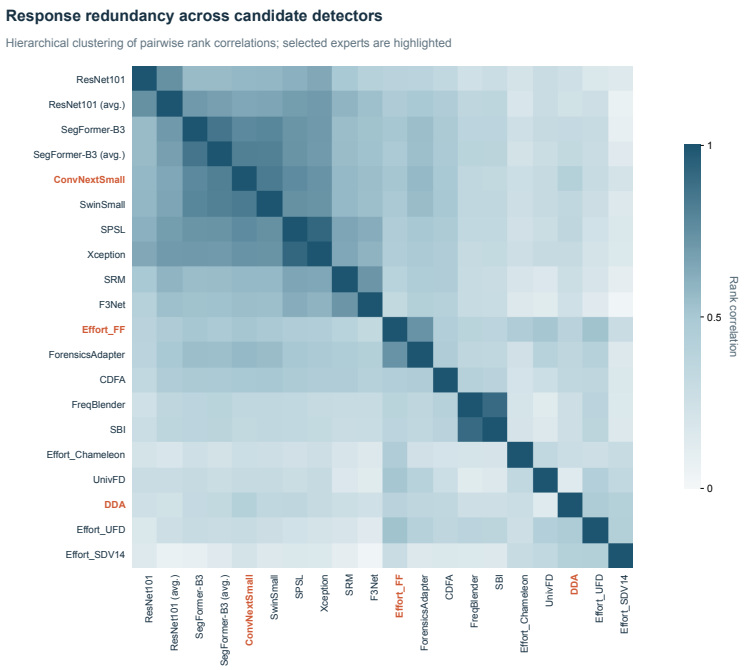

Figure 5: Pairwise rank correlations of the 20 detector responses on shared reference images. Detector order follows hierarchical clustering; the selected experts are highlighted.

A.5 Computational Cost

For the DINOv3 ViT-7B/16 and DINOv3 ViT-B/16 experiments, image features were extracted once and reused across selection methods, rounds, and seeds. The ScaleDF train/validation extraction records cover 5.94 million images for each backbone; the experiment-specific feature cache contains 5.93 million aligned rows from the available feature assets. We trained only the classifier during iterative selection. Table 9 separates feature extraction from the subsequent experiments.

Table 9: Approximate compute for the reported experiments. Extraction times sum the per-source durations recorded on NVIDIA H100 GPUs; experiment times are derived from run timestamps, except for CLIP/MetaCLIP, which report summed training-epoch durations. The latter two targets compute frozen image features online during training.

Stage	Scope	GPU hours
Candidate-detector reference scoring	20 detectors in total	~ 38
DINOv3 ViT-7B/16 feature extraction	5.94M ScaleDF images	~ 10.4
DINOv3 ViT-B/16 feature extraction	5.94M ScaleDF images	~ 1.2
DINOv3 ViT-7B/16 iterative experiments	18 configurations $\times$ 6 seeds	~ 10.6
DINOv3 ViT-7B/16 FF++ initialization	4 configurations $\times$ 6 seeds	~ 2.4
DINOv3 ViT-B/16 iterative experiments	3 configurations $\times$ 6 seeds	~ 1.1
CLIP ViT-L/14 training	30 final/initial models	~ 14.9
MetaCLIP ViT-B/32 training	30 final/initial models	~ 15.0
Approximate total	Listed stages	~ 94

A.6 Paired Selection Comparisons Across Initializations

Table 10 reports paired differences in the mean AUC across six external test sets at the final 250,000-sample budget. For each initialization, we compared matched training seeds and averaged the two AUC differences. All six seed-pair differences were positive in each of the three comparisons.

Table 10: Paired differences in external average AUC (percentage points) on DINOv3 ViT-7B/16. Each entry is the mean of two differences paired by initialization and training seed at 250,000 added samples. Boundary-error and confident-error denote the anchor-identification rule; selection trajectories can diverge after retraining. The final row summarizes the three initialization means with sample standard deviation.

Initialization	Bound. err. (diff. − score)	Conf. err. (diff. − score)	Dynamic − fixed
42	+3.89	+3.11	+2.21
43	+3.89	+3.21	+4.31
44	+3.74	+4.75	+1.49
Mean ± SD	$+3.84 \pm 0.09$	$+3.69 \pm 0.92$	$+2.67 \pm 1.47$

A.7 Full Statistics for Compact Result Tables

Unless noted otherwise, results below are AUC (%) reported as mean ± sample standard deviation across three initializations, each averaging two classifier-training seeds.

Table 11: Matching and anchor ablations after 250,000 additions on DINOv3 ViT-7B/16.

Configuration	Held-out B	External avg.
Candidate matching with boundary-error anchors		
score	93.9 ± 0.2	86.9 ± 0.5
Expert scores	94.2 ± 0.1	90.3 ± 0.5
diff.	94.5 ± 0.2	90.7 ± 0.5
Anchor identification and matching		
Conf. err. / score	93.5 ± 0.6	86.6 ± 0.8
Conf. err. / diff.	94.2 ± 0.2	90.3 ± 0.3
Bound. err. / score	93.9 ± 0.2	86.9 ± 0.5
Bound. err. / diff.	94.5 ± 0.2	90.7 ± 0.5

Table 12: Expert composition and iterative updating after 250,000 additions on DINOv3 ViT-7B/16.

Configuration	Held-out B	External avg.
Expert groups and score baseline		
Bound. err. / score	93.92 ± 0.17	86.89 ± 0.45
$K = 2$: ConvNextSmall + Effort_FF	94.16 ± 0.05	90.60 ± 0.41
$K = 2$: Effort_FF + DDA	94.42 ± 0.16	89.72 ± 0.66
$K = 2$: ConvNextSmall + DDA	94.21 ± 0.14	90.31 ± 0.44
$K = 3$: Effort_FF + DDA + SwinSmall	94.70 ± 0.31	90.49 ± 0.64
$K = 3$: ConvNextSmall + DDA + ForensicsAdapter	94.23 ± 0.19	90.87 ± 0.07
$K = 3$: ConvNextSmall + Effort_FF + DDA	94.51 ± 0.22	90.74 ± 0.49

Table 13: Initial training set and target-backbone comparisons with available standard deviations. Other backbone means are reported in Table 6.

Starting point	Selection method	Held-out B	External avg.
Initial training set (DINOv3 ViT-7B/16 target)			
Multi-source	Initial model	92.0 ± 0.2	85.0 ± 1.5
	Rand. / score	92.2 ± 0.2	86.6 ± 1.0
	Bound. err. / score	93.9 ± 0.2	86.9 ± 0.5
	Conf. err. / diff.	94.2 ± 0.2	90.3 ± 0.3
	Bound. err. / diff.	94.5 ± 0.2	90.7 ± 0.5
FF++	Initial model	80.9 ± 0.3	79.8 ± 0.7
	Rand. / score	90.1 ± 0.4	85.9 ± 0.7
	Bound. err. / score	93.2 ± 0.3	87.8 ± 0.4
	Conf. err. / diff.	94.4 ± 0.1	90.2 ± 0.3
	Bound. err. / diff.	94.2 ± 0.1	90.3 ± 0.3
Target backbone (multi-source initialization)			
DINOv3 ViT-B/16	Initial model	85.1 ± 0.1	75.8 ± 2.3
	Rand. / score	85.2 ± 0.3	77.5 ± 1.0
	Bound. err. / diff.	87.1 ± 0.4	84.5 ± 0.6
DINOv3 ViT-7B/16	Initial model	92.0 ± 0.2	85.0 ± 1.5
	Rand. / score	92.2 ± 0.2	86.6 ± 1.0
	Bound. err. / diff.	94.5 ± 0.2	90.7 ± 0.5