SyRC: Synthesis–Resampling Consistency for Deepfake Detection

Anonymous authors
Paper under double-blind review

Abstract

Online forgery synthesis diversifies deepfake training data, yet detectors trained on these examples can remain sensitive to downsampling and restoration. To complement forgery diversity with stability under resampling, we introduce Synthesis–Resampling Consistency (SyRC). During training, SyRC refreshes self-blended forgeries online and constrains the feature discrepancy between each augmented instance and its resampled view. By sharing the underlying image and base augmentation draw, the two views isolate resampling-induced changes for the consistency constraint. The additional operations are confined to training, preserving single-view inference. With GenD trained on FaceForensics++, SyRC improves detection on four target datasets. The mean video-level area under the receiver operating characteristic curve (AUC) increases from 89.80% to 93.59%. Matched comparisons also favor shared-base pairing over independent augmentation, supporting controlled within-instance consistency as a useful complement to synthetic forgery diversity.

1 Introduction

Deepfake detectors must recognize manipulated faces beyond the synthesis methods and image-processing conditions represented in their training data (Rössler et al., 2019; Li et al., 2020b; Dolhansky et al., 2020). These conditions affect different stages: synthesis creates a forgery, while subsequent resizing or compression changes its appearance. Figure 1(a–b) contrasts repeated use of a fixed forgery realization with appearance changes introduced after formation. Training for this setting therefore calls for both varied forgery examples and stable predictions under label-preserving processing.

Online synthesis supplies varied examples by generating fresh forgeries during training. The Self-Blended Images (SBI) procedure creates pseudo-fakes by transforming and blending pristine facial content (Shiohara & Yamasaki, 2022; Li et al., 2020a). Regenerating these blends across training visits changes the forgery realizations available to the detector. We examine whether this refresh also provides stability under resampling using the GenD detector (Yermakov et al., 2026). We use downsampling to 128×128 followed by restoration to the input size (DR-128) as a controlled resampling perturbation. On FaceForensics++ (FF++) validation data, DR-128 still changes fake–real orderings after online SBI refresh and resampling exposure (Figure 3(b)).

A direct response to this sensitivity is to expose the detector to resampled images during training. Task supervision encourages correct labels under both conditions, while a paired feature loss can explicitly constrain the difference between representations of the same instance. Building on augmentation-consistency learning (Hendrycks et al., 2020), we test feature consistency as a complement to resampling exposure.

Pair construction determines which changes this loss penalizes. Independently augmenting each view introduces differences from both augmentation randomness and resampling. We instead apply the base augmentation pipeline once and resample its output, so that both views share the same instance and augmentation draw. Synthesis–Resampling Consistency (SyRC) combines this shared-base construction with online SBI refresh, as summarized in Figure 1(c). Both real and fake instances receive paired consistency, while one randomly selected view supplies task supervision. All additional operations are confined to training, preserving single-view inference.

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.

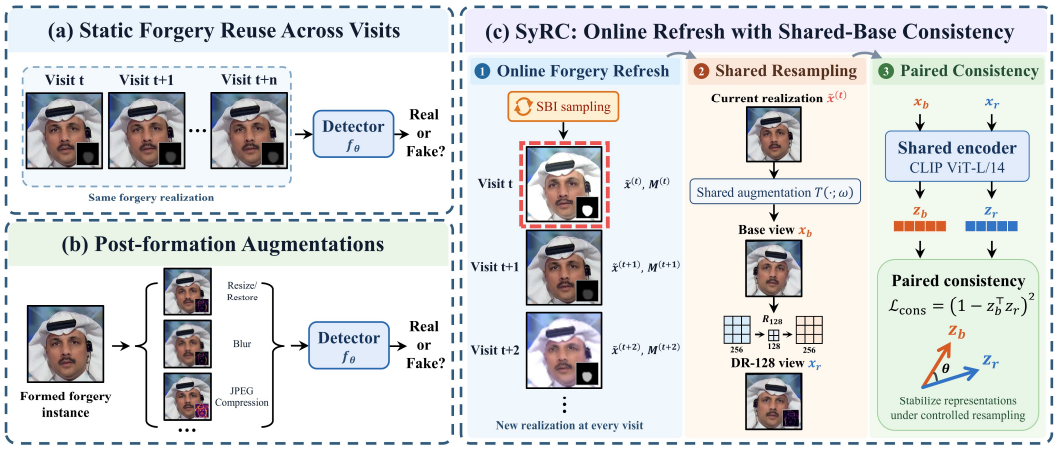

Figure 1: Training-instance variation addressed by SyRC. (a) Fixed replay reuses the same formed forgery across training visits. (b) Post-formation augmentation changes the appearance of the formed instance. (c) SyRC (1) refreshes selected SBI forgeries online, (2) constructs a shared-base/resampled pair, and (3) applies paired feature consistency. The grayscale inset denotes the SBI blending mask and is not an additional detector input.

The contributions are threefold:

- We introduce SyRC, combining forgery diversity with resampling consistency. Online refresh varies the training forgeries, while shared-base pairing keeps the instance and augmentation draw identical within each pair, focusing the feature constraint on resampling-induced changes.
- We show that feature consistency improves detection beyond resampling exposure alone across three source mixtures, including one using original FF++ fakes. Matched comparisons also favor online refresh over fixed replay and shared-base pairing over independent augmentation in mean target AUC. Source-validation diagnostics connect reduced decision-normal and video-score changes to a smaller net AUC loss under DR-128.
- SyRC improves cross-dataset detection with GenD and in an Effort transfer evaluation while preserving single-view inference. With FF++ training, mean video-level AUC across four target datasets rises from 89.80% to 93.59% on GenD, while the Effort gain averages 3.07 percentage points across three seeds.

2 Related Work

2.1 Foundation-Model Deepfake Detection

Recent deepfake detectors adapt pretrained visual representations through restricted parameter updates, token-level modeling, lightweight adapters, or vision–language supervision (Yan et al., 2025; Wang et al., 2026; Sun et al., 2026; Zhang et al., 2026; Zhu et al., 2026). These designs modify how pretrained features are updated, combined, or supervised for authenticity prediction.

We retain GenD's Contrastive Language–Image Pre-training (CLIP) detector and base objective (Radford et al., 2021; Yermakov et al., 2026). SyRC changes the training instances and their view relationships by refreshing selected synthetic forgeries and coupling each instance with a resampled view.

2.2 Synthetic Forgery Generation

Blending-based methods expand the range of training forgeries using pristine images. Face X-ray emphasizes blending boundaries, while SBI generates pseudo-fakes by transforming and blending

facial content (Li et al., 2020a; Shiohara & Yamasaki, 2022). Learning Self-Consistency uses online synthesis to supervise consistency within an image (Zhao et al., 2021).

We adopt SBI as the synthesis primitive and compare fixed replay with regeneration across training visits. The resulting instances are paired with their resampled views, so the consistency constraint relates two views of the same instance.

2.3 Resampling and View Consistency

Post-processing can alter the visual evidence available to a detector. DeeperForensics studies diverse real-world perturbations, while GenD evaluates blur, compression, and downsampling followed by restoration (Jiang et al., 2020; Yermakov et al., 2026). Consistency learning uses transformed views as a training signal by relating their predictions or representations. AugMix, for example, shares the initial crop and flip before constructing and regularizing multiple views (Hendrycks et al., 2020).

In SyRC, the views share the same base augmentation draw: the second view is obtained by downsampling and restoring the already augmented instance. Comparison with independently augmented pairing tests the role of this shared augmentation randomness.

3 Method

To learn from diverse forgeries while reducing sensitivity to resampling, SyRC combines online SBI refresh with shared-base feature consistency. Figure 2 summarizes how these components interact during training and how the resulting detector operates at inference.

3.1 Online Forgery Refresh

Online refresh varies forgery content within existing training entries. Given an entry with image x and real/fake label y, we define $\tilde{x}$ as the image used for subsequent view construction:

$$\tilde{x} = \begin{cases} G_{\mathrm{SBI}}(x_{\mathrm{source\ real}}; \xi), & \text{selected fake entry,} \\ x, & \text{otherwise.} \end{cases} \tag{1}$$

Here, $x_{\mathrm{source\ real}}$ is a fixed pristine FF++ training frame from the selected fake entry's paired background video. The SBI generator G_{SBI} transforms a copy of this face and blends it into the source face (Shiohara & Yamasaki, 2022). We resample the synthesis randomness ξ on every selected visit; real and unselected fake entries retain their original images. The label y remains unchanged.

We replace approximately 20% of fake entries ($r = 0.2$) using a method-stratified five-epoch schedule, selecting each fake entry approximately once per cycle. Assuming equal exposure to the four original FF++ manipulation families, replacement leaves each with a share $(1 - r)/4$ of fake instances and SBI with r. We set $r = 1/5 = 0.2$ so that SBI and each original manipulation family contribute the same expected share of fake instances. Replacement preserves the per-epoch sample count, optimizer-step count, and real/fake ratio. Fixed replay reuses previously generated SBI content; Section 4.3 compares replay and online refresh. Refresh diversifies the synthetic content seen across visits, but does not constrain how those images respond to resampling.

3.2 Shared-Base View Construction

To isolate the effect of resampling, we apply the base augmentation pipeline T to each real or fake instance $\tilde{x}$, with random parameters ω, to obtain the base view x_b. We then derive the resampled view x_r directly from x_b:

$$\begin{aligned} x_b &= T(\tilde{x}; \omega), \\ x_r &= R_{128}(x_b) = U_{256}^{\mathrm{bicubic}}\big(D_{128}^{\mathrm{box}}(x_b)\big)\,. \end{aligned} \tag{2}$$

Here, R_{128} denotes downsampling and restoration (DR-128): D_{128}^{box} downsamples the 256×256 base view to 128×128 using BOX interpolation, and $U_{256}^{\mathrm{bicubic}}$ restores it to 256×256 using bicubic interpolation. Both views retain the label of $\tilde{x}$ and undergo the same pretrained CLIP preprocessing

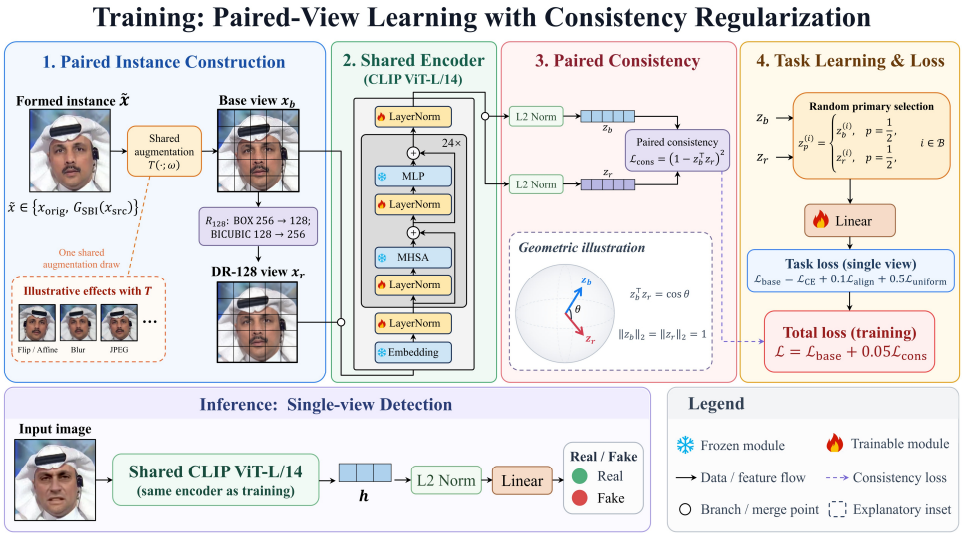

Figure 2: Overview of SyRC training and inference. The formed instance $\tilde{x}$, either original or an online-generated SBI, is augmented once to form x_b, which DR-128 converts into the paired view x_r. The shared encoder produces normalized features for consistency, while one randomly selected view supplies the task loss. In GenD, only the LayerNorm affine parameters and linear classifier are updated; inference processes one view per frame. MHSA and MLP denote multi-head self-attention and multilayer perceptron.

to produce 224×224 encoder inputs. Appendix A specifies the augmentation and preprocessing pipelines.

Because x_r is derived from x_b, the pair shares the underlying image and base augmentation draw; only resampling differs, isolating this change for the consistency objective. Independent pairing instead uses a separate base augmentation draw for each view; Section 4.5 compares the two constructions.

3.3 Consistency and Task Supervision

Task supervision learns the real/fake distinction from one randomly selected view per instance. Paired consistency constrains resampling-induced representation changes using both views.

Shared representation. We use GenD's adaptation of CLIP ViT-L/14, a large Vision Transformer with 14-pixel patches (Yermakov et al., 2026). Both views pass through the same encoder h_θ, with CLIP preprocessing included in this mapping. Its output is the classification-token (`[CLS]`) feature after the final LayerNorm. For each nonzero output, we use the unit-normalized feature $z_j = h_\theta(x_j)/\|h_\theta(x_j)\|_2$, where $j \in \{b, r\}$. Only LayerNorm affine parameters and the linear real/fake classifier are updated.

Paired consistency. We penalize the squared cosine gap between the two normalized features:

$$\mathcal{L}_{\text{cons}} = \left(1 - z_b^\top z_r\right)^2. \tag{3}$$

Squaring gives larger cosine gaps greater relative weight, with a diminishing slope as features align. With shared-base pairing, this loss penalizes the isolated resampling change.

Single-view task supervision. The inherited GenD objective combines cross-entropy (CE), alignment, and uniformity (Wang & Isola, 2020; Yermakov et al., 2026):

$$\mathcal{L}_{\text{base}} = \mathcal{L}_{\text{CE}} + 0.1\,\mathcal{L}_{\text{align}} + 0.5\,\mathcal{L}_{\text{uniform}}.$$

For each instance i in microbatch $\mathcal{B}$, we choose the base or resampled view with equal probability:

$$x^{(i)}_{\text{primary}} \sim \text{Uniform}\left(\{x^{(i)}_b, x^{(i)}_r\}\right), \qquad i \in \mathcal{B}. \tag{4}$$

The full microbatch objective is

$$\mathcal{L} = \mathcal{L}_{\text{base}}(\mathcal{B}_{\text{primary}}) + 0.05\,\frac{1}{|\mathcal{B}|}\sum_{i\in\mathcal{B}}\mathcal{L}_{\text{cons}}^{(i)}, \tag{5}$$

Here, $\mathcal{B}_{\text{primary}}$ contains one selected view and label per instance. All three task-loss components use this microbatch; consistency is averaged over its base–resampled pairs. Selecting one primary view preserves the number of task-supervised instances in each microbatch. GenD's alignment loss brings together features from same-class instances, whereas paired consistency relates two views of the same instance. The latter directly constrains resampling-induced changes while retaining the inherited class-discrimination objective.

At inference, the detector encodes one view per frame, applies the linear head, and averages frame fake probabilities into a video score, without online synthesis or paired-view computation.

3.4 Resampling Stability of Video Rankings

The shared linear classifier connects paired feature changes to the video scores used for evaluation. At a fixed checkpoint, let $v = w_f - w_r$ be the difference between the fake and real classifier weights. For the same n_V frames of video V, let $S_b(V)$ and $S_r(V)$ denote the mean fake probabilities before and after resampling, and let $\ell_i = (1 - z_{b,i}^{\top} z_{r,i})^2$ be the paired consistency loss for frame i. Unit-normalized features and the Lipschitz property of the sigmoid give

$$|S_r(V) - S_b(V)| \le \frac{\|v\|_2}{2\sqrt{2}}\left(\frac{1}{n_V}\sum_{i=1}^{n_V}\ell_i\right)^{1/4} \equiv B_V. \tag{6}$$

Here $\ell_i = \mathcal{L}_{\text{cons}}^{(i)}$ is the framewise consistency loss used in training, while B_V and the ranking condition below are analytical quantities rather than additional loss terms. Appendix H provides the complete derivation. Video-level AUC depends on the ordering of fake–real video pairs. For a fake video V^+ and a real video V^-, the clean ordering is therefore guaranteed to remain unchanged when

$$S_b(V^+) - S_b(V^-) > B_{V^+} + B_{V^-}. \tag{7}$$

This sufficient condition certifies pairs with margins larger than their combined score-change bounds; it does not determine the behavior of the remaining pairs. At the dataset level, the AUC response depends on both initially correct pre-resampling orderings that become reversed and initially incorrect orderings that are recovered.

Task supervision and paired consistency act on different parts of this relation. Task supervision shapes the pre-resampling fake–real separation, whereas paired consistency penalizes the feature change entering B_V; DR exposure alone supervises both input conditions without explicitly coupling their representations. Shared-base pairing keeps the paired difference specific to resampling rather than to a second augmentation draw. Section 4.4 tests these distinctions through exposure-only and pairing controls and examines the corresponding feature, score, and ranking changes; Appendix H derives the pairwise and aggregate relations.

4 Experiments

We test whether SyRC improves cross-dataset detection, whether refresh, consistency, and pairing contribute under matched controls, and whether consistency changes the response to resampling.

4.1 Experimental Setup

Datasets and evaluation. We train on the c23 compression version of FF++ (Rössler et al., 2019) and select checkpoints by FF++ validation video-level AUC. The four target datasets are Celeb-DF versions 1 and 2 (CDF-v1/v2), the DeepFake Detection Challenge Preview dataset (DFDCP), and the DeepFake Detection Challenge dataset (DFDC) (Li et al., 2020b; Dolhansky et al., 2019; 2020). All internal comparisons use the same DeepfakeBench-derived evaluation lists (Yan et al., 2023b), comprising $168{,}894$ frames from $5{,}976$ videos. We average frame fake probabilities within

each video, compute video-level AUC for each target, and report the equal-weight mean of the four AUCs. Unless explicitly identified as multi-seed results, ablations use designated single runs; per-target comparisons use seed 42.

Training settings. All GenD configurations use AdamW for 20 epochs and update only the LayerNorm affine parameters and classifier. Losses are computed within 16-sample microbatches, with eight gradient-accumulation steps per optimizer update (effective batch size 128). SyRC replaces approximately 20% of fake entries and uses DR-128 with a consistency weight of 0.05. We select the resampling resolution and consistency weight by FF++ validation video-level AUC, then keep them fixed for all reported target comparisons.

Resampling diagnostics. We test whether paired consistency reduces the response to DR-128 at three levels: decision-relevant features, video scores, and fake–real rankings. Using $v = w_f - w_r$ from Section 3.4, define $\hat{v} = v/\|v\|_2$. For normalized base and resampled features, the decision-normal displacement is

$$\Delta_R = \left|\hat{v}^\top (z_r - z_b)\right|. \tag{8}$$

A smaller Δ_R means that resampling changes the feature less along the classifier's decision direction. At the video level, $|S_r(V) - S_b(V)|$ measures score change. The strict adverse reversal rate counts all fake–real pairs that are correct on clean inputs but reversed after DR-128; the recovered-ordering rate counts pairs that change in the opposite direction. Their difference equals $\mathrm{AUC}_{\mathrm{clean}} - \mathrm{AUC}_{\mathrm{DR\text{-}128}}$. All diagnostics use the same DR-128 cache of 700 FF++ validation videos (140 real and 560 fake). We first average frame-level displacement within each video; displacement and video-score change are then averaged equally across the real and fake classes. The comparison matches online SBI, task-supervised sample count, and primary-view DR exposure: the exposure-only checkpoint encodes one sampled view, whereas full SyRC additionally encodes its paired view and applies consistency.

4.2 Cross-Dataset Generalization

SyRC improves the four-target mean video-level AUC by 3.79 percentage points on GenD and by 3.07 points on average across three Effort seeds under the shared source and target protocol (Table 1). Both detectors improve on all four targets, extending the observed benefit beyond the main GenD implementation. Table 1 also reports the per-target comparisons, while Appendix A provides the full seed summaries.

Table 1: Internal baseline–SyRC comparisons. Top: three-seed mean ± SD of four-target mean video-level AUC (%) and gains in percentage points; bottom: seed-42 per-target AUC.

Detector	Baseline	+ SyRC	Gain
GenD	89.80 ± 0.12	93.59 ± 0.03	+3.79
Effort	90.22 ± 0.07	93.29 ± 0.32	+3.07

Detector	Configuration	CDF-v1	CDF-v2	DFDCP	DFDC	Avg.
GenD	Baseline	92.53	93.02	88.14	85.21	89.73
	+ SyRC	**97.20**	**96.16**	**93.38**	**87.69**	**93.61**
Effort	Baseline	91.68	94.30	89.18	85.73	90.22
	+ SyRC	**96.69**	**95.62**	**93.17**	**88.91**	**93.60**

Table 2 places the seed-42 GenD result alongside published detectors. SyRC has the highest mean among the listed methods reporting all four targets, although DFF-Adapter remains stronger on DFDC. The following internal comparisons examine which training choices contribute to the improvement.

4.3 Effect of Online Forgery Refresh

To distinguish regeneration from reusing synthetic content, we compare fixed-SBI replay with online-SBI refresh while matching SBI exposure and training budget. Across three seeds, online

Table 2: Cross-dataset video-level AUC (%) after FF++ training. Prior results are published; the SBI row is from Table 1 of DFF-Adapter (Zhang et al., 2026). Avg. equally weights the four targets. Dashes indicate unavailable results; bold and underlining mark the highest and second-highest values.

Method	Venue	Unseen Target Datasets				
		CDF-v1	CDF-v2	DFDCP	DFDC	Avg.
Xception (Rössler et al., 2019)	ICCV'19	80.98	81.65	69.90	–	–
Face X-ray (Li et al., 2020a)	CVPR'20	80.58	79.50	80.92	–	–
SPSL (Liu et al., 2021)	CVPR'21	85.02	79.86	75.86	66.16	76.73
SBI (Shiohara & Yamasaki, 2022)	CVPR'22	93.44	93.82	90.95	74.47	88.17
UCF (Yan et al., 2023a)	ICCV'23	86.08	83.73	80.50	75.11	81.36
ProDet (Cheng et al., 2024)	NeurIPS'24	94.48	92.62	82.83	71.52	85.36
UDD (Fu et al., 2025)	AAAI'25	–	93.13	88.11	81.21	–
Forensics Adapter (Cui et al., 2025)	CVPR'25	–	95.70	92.90	87.20	–
Effort (Yan et al., 2025)	ICML'25	–	95.60	90.90	84.30	–
DeepShield (Cai et al., 2025)	ICCV'25	–	92.20	<u>93.20</u>	82.80	–
GenD (CLIP) (Yermakov et al., 2026)	WACV'26	–	<u>96.00</u>	–	87.10	–
DFF-Adapter (Zhang et al., 2026)	AAAI'26	<u>96.14</u>	95.26	91.57	**89.96**	<u>93.23</u>
DFD-HR (Sun et al., 2026)	CVPR'26	–	<u>96.00</u>	90.10	86.50	–
Revisiting FFD (Guo et al., 2026)	TPAMI'26	90.46	93.02	87.16	84.90	88.89
SyRC	–	**97.20**	**96.16**	**93.38**	<u>87.69</u>	**93.61**

Table 3: Fixed-SBI replay versus matched online-SBI refresh at seed 42. Values are video-level AUC (%); Avg. is the four-target mean. Differences are in percentage points.

Replay and matched refresh policy	Unseen Target Datasets				
	CDF-v1	CDF-v2	DFDCP	DFDC	Avg.
Fixed-SBI replay	95.824	95.513	**89.770**	85.841	91.737
Matched online-SBI refresh	**96.477**	**95.919**	89.707	**86.065**	**92.042**
Online − Fixed	+0.653	+0.406	−0.063	+0.224	+0.305

refresh improves the four-target mean video-level AUC by 0.305–0.603 percentage points, with a mean gain of 0.492 points (Figure 3(a)). The seed-42 breakdown improves on CDF-v1, CDF-v2, and DFDC but decreases by 0.063 points on DFDCP (Table 3), showing a consistent average benefit across seeds but not a uniform per-dataset effect; Appendix C reports the absolute seed results and the auxiliary SBI–BI comparison.

4.4 Consistency across Source Mixtures

The refresh comparison isolates regeneration, but it does not show whether consistency contributes beyond exposure to resampled images. We therefore compare training without resampling, resampling exposure, and paired consistency within each source mixture. We first examine the main online-SBI setting across three seeds, and then repeat the exposure–consistency comparison with original FF++ fakes and fixed-SBI replay to test whether the effect depends on online synthesis.

Adding consistency after DR exposure improves mean target AUC in all three online-SBI runs, with gains of 0.67–0.87 percentage points and an average gain of 0.80 points (Table 4). The gain remains positive with original FF++ fakes and fixed-SBI replay, where consistency adds 1.38 and 0.58 points, respectively. Because these comparisons hold the source mixture fixed, they show that the benefit over exposure alone does not depend on online synthesis.

The replay–refresh comparison isolates regeneration from synthetic-content reuse, whereas the within-mixture comparisons isolate paired consistency from DR exposure. Figure 3(a) summarizes the refresh gain. Panels (b–c) then compare exposure-only training with full SyRC through their DR-128 ranking changes and decision-normal displacement. We next connect these diagnostics to the score response.

Table 4: Paired consistency beyond resampling exposure. Values are four-target mean video-level AUC (%); online-SBI rows are matched runs, and the final two rows are designated source-mixture controls.

Setting	No DR	+ DR exposure	+ Consistency	Gain
Online SBI, seed 0	91.85	92.73	93.59	+0.86
Online SBI, seed 42	92.04	92.74	93.61	+0.87
Online SBI, seed 2026	91.98	92.89	93.56	+0.67
Online SBI, mean ± SD	91.96 ± 0.10	92.79 ± 0.09	$\mathbf{93.59 \pm 0.03}$	+0.80
Original FF++ fakes	89.73	90.71	92.09	+1.38
Fixed-SBI replay	91.74	92.54	93.12	+0.58

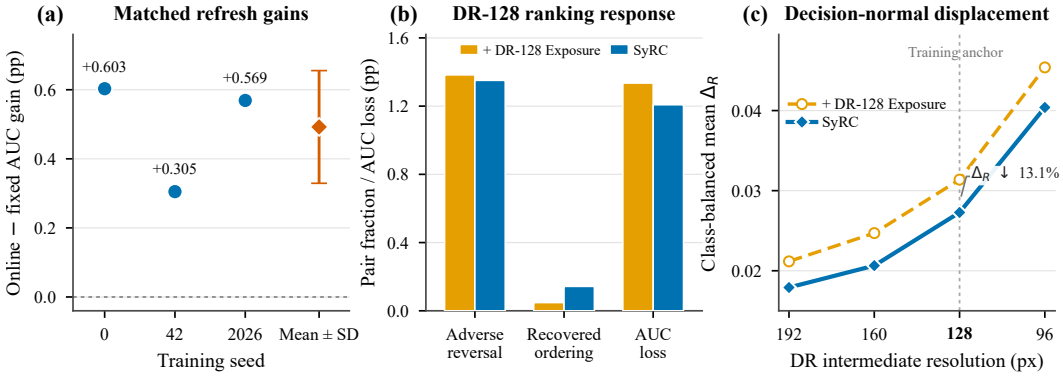

Figure 3: Matched refresh gains and resampling diagnostics. (a) Four-target mean video-level AUC gain of online-SBI refresh over fixed replay across three seeds (diamond and error bar: mean ± sample SD). (b) Strict adverse reversals, recovered orderings, and their net AUC loss under DR-128 for exposure-only training and SyRC. (c) Class-balanced mean decision-normal displacement Δ_R across resampling resolutions. Panels (b–c) use seed 42; the dotted line in (c) marks the 128-pixel training condition.

Resampling-induced score and ranking changes. The target-AUC gain over exposure does not show how consistency changes the response to resampling. We therefore follow Section 3.4 from decision-normal displacement to video-score change and ranking changes. On the matched validation cache, the class-balanced mean absolute video-score change decreases from 0.0453 to 0.0382 (15.8%), with reductions for both real (14.8%) and fake (18.6%) videos. The adverse reversal rate decreases slightly from 1.383% to 1.351%, while the recovered-ordering rate increases from 0.048% to 0.143%; together, these changes reduce AUC loss from 1.335 to 1.208 percentage points (Figure 3(b)). Together, panels (b–c) connect the measured feature and score changes to the observed ranking response. We next test whether shared-base pairing isolates resampling from additional augmentation variation.

4.5 Shared-Base Pairing Analysis

Pair construction determines whether the consistency loss isolates resampling or also absorbs a second augmentation draw: shared-base pairing derives the resampled view from an already augmented base view, whereas independent pairing augments the two branches separately. We keep the source mixture, resampling operation, consistency objective, loss weight, and training budget fixed. At the fixed online-formation checkpoint, we measure pair discrepancy by the unsquared cosine gap $g(a, b) = 1 - z(a)^\top z(b)$ between normalized features. Because $\mathcal{L}_{\text{cons}} = g(z_b, z_r)^2$, the gap indicates how much variation each construction presents to the consistency objective.

Shared-base pairing improves the four-target mean AUC in all three matched seeds, with gains of 0.36–0.57 percentage points and an average gain of 0.46 points (Table 5, top). Because the runs differ only in pair construction, this result favors sharing the base augmentation under the same consistency objective.

Table 5: Shared-base versus independent pairing. Target values are four-dataset mean video-level AUC (%) in matched training runs; cosine gaps are class-balanced diagnostics at the fixed seed-42 online-formation checkpoint.

Seed	Independent	Shared-base	Shared − Independent
0	93.02	93.59	+0.57
42	93.25	93.61	+0.36
2026	93.11	93.56	+0.45
Mean ± SD	93.13 ± 0.12	93.59 ± 0.03	+0.46
Fixed-checkpoint pair			Cosine gap
Two independently augmented views			0.5054 ± 0.0010
Shared-base pair			$\mathbf{0.4058 \pm 0.0008}$
Independent pair			0.5723 ± 0.0007

At the fixed online-formation checkpoint, shared-base pairs also have a smaller cosine gap than independent pairs in all three augmentation draws, while the independently augmented reference lies between them (Table 5, bottom). This ordering is consistent with an independent second augmentation draw adding variation beyond resampling. The cosine-gap diagnostic compares the discrepancy presented by each construction; the matched runs test its effect on target AUC. Appendix D gives the constructions and aggregation.

As a boundary control, Appendix F replaces DR-128 with Gaussian blur; the tested replacement reduces the four-target mean AUC from 93.61% to 92.06%.

4.6 Training Cost and Inference

Half pairing preserves primary-view task supervision for every instance but constructs the paired resampled view for only half, reducing the average number of encoded views from 2.0 to 1.5. On an NVIDIA RTX 3060 Laptop GPU, this reduces GPU time by 24.2% and peak memory by 19.6%, while increasing throughput by 22.0%, for a 0.08-point decrease in mean target AUC. Both settings retain single-view inference. Appendix A reports the complete profiling results and protocol.

5 Discussion

Matched controls separate complementary gains from online regeneration and paired consistency beyond DR exposure. Shared-base pairing further outperforms independent pairing, while its smaller fixed-checkpoint cosine gap is consistent with excluding a second augmentation draw.

DFDC remains the weakest target for both detectors, and DFF-Adapter remains stronger there, so the average gain does not imply uniform transfer. Applying SyRC beyond facial forgeries would require view transformations that alter image quality without changing the real/fake label.

The analysis bounds video-score changes and certifies ranking stability when the pre-resampling margin exceeds the combined score-change bounds; other pairs depend on the signed score changes. Under DR-128, SyRC reduces decision-normal displacement and video-score change, while fewer adverse reversals and more recovered orderings yield a smaller net AUC loss.

6 Conclusion

SyRC combines online forgery refresh with consistency between views of the same augmented instance. Matched controls support regeneration and shared-base pairing, and the full procedure improves cross-dataset detection across three GenD training seeds and three Effort transfer runs. Together, these results support resampling consistency as a complement to forgery diversity, with single-view inference preserved.

AI USE STATEMENT

Generative artificial intelligence (AI) tools were used for literature discovery, research ideation, translation, language editing, and assistance with figure design. The authors reviewed and finalized all AI-assisted material.

ETHICS STATEMENT

SyRC uses synthetic facial blends for training and is evaluated on established research benchmarks. Deepfake detection can support media authentication and forensic analysis, but detector scores alone do not establish authenticity. Performance may vary across data sources and demographic groups. Any deployment should include human review, communicate this uncertainty, and comply with dataset licenses and consent conditions.

REPRODUCIBILITY STATEMENT

Section 3 defines sample replacement, paired-view construction, view selection, training objectives, and inference. Section 4 specifies checkpoint selection, evaluation, and diagnostic aggregation, and identifies single-run and multi-seed results. Appendix A provides the augmentation and preprocessing pipelines, source sampling, microbatch loss computation, and per-seed results.

REFERENCES

Yinqi Cai, Jichang Li, Zhaolun Li, Weikai Chen, Rushi Lan, Xi Xie, Xiaonan Luo, and Guanbin Li. Deepshield: Fortifying deepfake video detection with local and global forgery analysis. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 12524–12534, 2025.

Jikang Cheng, Zhiyuan Yan, Ying Zhang, Yuhao Luo, Zhongyuan Wang, and Chen Li. Can we leave deepfake data behind in training deepfake detector? In *Advances in Neural Information Processing Systems*, volume 37, pp. 21979–21998, 2024.

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, Russ Howes, Ben Pflaum, Nicole Baram, and Cristian Canton Ferrer. The deepfake detection challenge preview dataset. *arXiv preprint arXiv:1910.08854*, 2019.

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

Xinghe Fu, Zhiyuan Yan, Taiping Yao, Shen Chen, and Xi Li. Exploring unbiased deepfake detection via token-level shuffling and mixing. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 3040–3048, 2025.

Z. Guo, Y. Liu, J. Zhang, H. Zheng, and S. Shan. Revisiting face forgery detection: From facial representation to forgery detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 48(8):9228–9244, 2026.

Dan Hendrycks, Norman Mu, Ekin D. Cubuk, Barret Zoph, Justin Gilmer, and Balaji Lakshminarayanan. Augmix: A simple data processing method to improve robustness and uncertainty. In *International Conference on Learning Representations*, 2020.

Liming Jiang, Ren Li, Wayne Wu, Chen Qian, and Chen Change Loy. Deeperforensics-1.0: A large-scale dataset for real-world face forgery detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2889–2898, 2020.

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, 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.

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.

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

Andreas Rössler, 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.

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.

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.

Changshuo Wang, Jiangming Wang, Ke-Yue Zhang, Taiping Yao, Shunli Wang, Shouhong Ding, Ran Yi, and Lizhuang Ma. Beyond [cls] token: Query-driven token-level forgery purification for generalizable deepfake detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 42922–42931, 2026.

Tongzhou Wang and Phillip Isola. Understanding contrastive representation learning through alignment and uniformity on the hypersphere. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 9929–9939, 2020.

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. In *Advances in Neural Information Processing Systems*, volume 36, pp. 4534–4565, 2023b.

Zhiyuan Yan, Jiangming Wang, Zhendong Wang, Peng Jin, Ke-Yue Zhang, Shen Chen, Taiping Yao, Shouhong Ding, Baoyuan Wu, and Li Yuan. Effort: Efficient orthogonal modeling for generalizable ai-generated image detection. In *Proceedings of the International Conference on Machine Learning*, 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.

Tianxiang Zhang, Peipeng Yu, Zhihua Xia, Longchen Dai, Xiaoyu Zhou, and Hui Gao. Fine-grained dino tuning with dual supervision for face forgery detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pp. 12780–12788, 2026.

Tianchen Zhao, Xiang Xu, Mingze Xu, Hui Ding, Yuanjun Xiong, and Wei Xia. Learning self-consistency for deepfake detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021.

Jiawen Zhu, Yunqi Miao, Xueyi Zhang, Jiankang Deng, and Guansong Pang. Unleashing vision-language semantics for deepfake video detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 42953–42963, 2026.

A Implementation and Evaluation Details

For the main GenD implementation, we train for 20 epochs using AdamW with zero weight decay. During the first epoch, the learning rate increases linearly from 10^{-5} to 3×10^{-4}. It then follows cosine annealing with warm restarts, using a 10-epoch restart period and a minimum learning rate of 10^{-5}. We use bfloat16 (BF16) mixed precision and accumulate gradients over eight 16-sample microbatches, giving an effective batch size of 128. The five-epoch SBI replacement cycle is independent of the learning-rate schedule.

For each real or fake instance, one of its base and resampled views is selected with equal probability for task supervision. Following GenD (Yermakov et al., 2026), we compute alignment and uniformity from the unit-normalized primary-view features within each microbatch. Consistency is averaged over the corresponding base–resampled feature pairs from the shared encoder. Gradients are accumulated across eight microbatches before an optimizer step; the losses are not recomputed over the effective batch.

Table 6: Additional implementation settings used by the GenD experiments.

Setting	Specification
Base augmentation pipeline T	256×256 PIL face $\rightarrow$ horizontal flip ($p = 0.5$) $\rightarrow$ affine (rotation $\pm10^\circ$, translation $\pm10\%$, scale $[0.9, 1.1]$) $\rightarrow$ Gaussian blur ($p = 0.1$, 7×7, $\sigma \in [0.1, 2.0]$) $\rightarrow$ brightness/contrast jitter ($[0.9, 1.1]$) $\rightarrow$ JPEG (quality 40–100).
Post-resampling preprocessing	DR operates on PIL images. Both views then pass through the pretrained CLIP processor: bicubic resizing to a short side of 224, a 224×224 center crop, rescaling, and CLIP normalization before ViT-L/14.
Source sampling	We use DeepfakeBench 256×256 face crops. Each selected fake entry uses a fixed pristine source: the nearest-index FF++ training-real frame from its paired background video. Each fake entry is selected approximately once per five-epoch cycle. Online refresh resamples only SBI formation randomness; dataset entries and labels are preserved.
Loss batch scope	Losses are computed within each 16-sample microbatch. Eight gradient-accumulation steps give an effective batch size of 128.
Training settings	SBI replacement rate $r = 0.2$, fixed DR-128, and consistency weight 0.05. Under equal exposure to the four original FF++ manipulation families, each retains a fake-instance share $(1 - r)/4$ and SBI occupies r; equating them gives $r = 1/5$.
Training-seed summaries	Seeds are 0, 42, and 2026. For multi-seed training results, $\pm$ denotes the sample standard deviation across these runs.

Target evaluation lists. All internal comparisons use the target frames and videos listed in Table 7. For each video, we average the fake probabilities of its evaluated frames. We then compute video-level AUC separately for each target dataset and average the four AUCs with equal weight. Dataset size therefore does not determine its contribution to the reported four-target mean.

Three-seed summary. Table 8 reports results for seeds 0, 42, and 2026 under the same training budget, source-validation checkpoint-selection rule, and target evaluation lists. It compares the baseline, online formation, resampling exposure, independent pairing, and full shared-base pairing. Each entry is the equal-weight mean of the four target-dataset video-level AUCs.

Table 7: Numbers of evaluated frames and videos in the DeepfakeBench-derived target lists shared by all internal comparisons.

Dataset	Frames	Videos
CDF-v1	3,136	100
CDF-v2	16,420	518
DFDCP	17,222	654
DFDC	132,116	4,704
Total	168,894	5,976

Table 8: Four-target mean video-level AUC (%) for five training configurations across three seeds. The final row reports the mean and sample standard deviation across training seeds.

Seed	Baseline	Online formation	Exposure only	Independent pair	Shared-base full
0	89.94	91.85	92.73	93.02	93.59
42	89.73	92.04	92.74	93.25	93.61
2026	89.74	91.98	92.89	93.11	93.56
Mean $\pm$ SD	89.80 ± 0.12	91.96 ± 0.10	92.79 ± 0.09	93.13 ± 0.12	93.59 ± 0.03

Partial pairing implementation details. With half pairing, every instance receives primary-view task supervision, and half of the instances also contribute a paired view to the consistency objective. The consistency loss is averaged over paired instances. We profile efficiency using synchronized GPU timing with 5 warm-up microbatches followed by 50 measured microbatches, excluding validation and checkpoint I/O.

Table 9: Training-cost tradeoff on an NVIDIA RTX 3060 Laptop GPU. Runtime, memory, and throughput are normalized to full pairing (1.000); half-pairing ratios are derived from the reported relative changes. AUC is the four-target video-level mean (%) in designated runs.

Metric	Full pairing	Half pairing
Encoded views / training instance	2.0	1.5
Relative GPU time $\downarrow$	1.000	0.758
Relative batch time $\downarrow$	1.000	0.820
Relative peak memory $\downarrow$	1.000	0.804
Relative throughput $\uparrow$	1.000	1.220
Mean target AUC $\uparrow$	93.61	93.53
Encoder passes / frame at inference	1	1

Effort application. We apply SyRC to Effort while retaining its CLIP ViT-L/14 backbone, rank-one SVD residual adaptation, linear classifier, and cross-entropy objective, and use the same 20% five-epoch online-SBI refresh schedule. After Effort's base augmentation, we derive the paired view using fixed DR-128 and apply squared-cosine consistency to the normalized 1024-dimensional CLIP pooler features; one randomly selected view supplies the classification objective. We train for 10 epochs with an effective batch size of 32 and select the checkpoint by FF++ validation video-level AUC. Inference retains Effort's original single-view path, with no paired view or additional parameters.

B Training-Setting Development

Development runs considered fixed intermediate resolutions in $\{96, 128, 160, 192\}$ and a configuration that sampled from the same set during training. We selected fixed DR-128 using FF++ validation video-level AUC and then held it constant for the reported target comparisons and displacement analyses. Its four-target mean AUC is 93.61%, compared with approximately 93.60% for

Table 10: Three-seed Effort transfer results. Values are equal-weight four-target mean video-level AUC (%); gains are in percentage points.

Seed	Baseline	+ SyRC	Gain
0	90.16	92.95	+2.79
42	90.22	93.60	+3.38
2026	90.29	93.32	+3.03
Mean $\pm$ SD	90.22 ± 0.07	93.29 ± 0.32	+3.07

the earlier multi-strength configuration. The single fixed resolution therefore provides a common intervention for the view-construction controls and a common anchor for the displacement analysis.

We selected the consistency weight by the same source-validation metric. After evaluating stronger settings, including 0.1, we fixed the weight at 0.05 for full, independent, and parallel pairing. The SBI replacement rate is determined separately by the equal-exposure rule in Section 3.

C Formation Details

C.1 Matched Replay–Refresh Results

Table 11 gives the absolute AUCs behind the matched refresh gains in Figure 3(a). Its matched online-SBI checkpoints are separate runs trained for the replay–refresh comparison rather than the online-formation checkpoints in Table 8.

Table 11: Fixed-SBI replay versus online-SBI refresh in the matched three-seed comparison. AUC values are equal-weight four-target video-level means (%); differences are in percentage points.

Seed	Fixed-SBI	Matched online-SBI	Online $-$ Fixed
0	91.756	92.359	+0.603
42	91.737	92.042	+0.305
2026	91.740	92.309	+0.569
Mean	91.744	92.237	+0.492

C.2 Choice of Synthesis Primitive

With downstream resampling and consistency fixed, SBI achieves 93.61% mean AUC versus 90.51% for Blended Images (BI), with gains on all four targets (Table 12). This comparison supports the choice of SBI within the tested pipeline; the replay–refresh comparison addresses how often its realizations are regenerated.

Table 12: Synthesis primitives with downstream resampling and consistency fixed. Values are video-level AUC (%); Avg. is the four-target mean.

Primitive	CDF-v1	CDF-v2	DFDCP	DFDC	Avg.
BI	96.33	93.23	89.46	83.00	90.51
SBI	**97.20**	**96.16**	**93.38**	**87.69**	**93.61**

D View Construction and Feature-Gap Diagnostics

This appendix details the pairing controls and source-validation diagnostics in Section 4.5.

Pairing controls. We omit ω below; repeated $T(\tilde{x})$ denotes the same base view, while T_1 and T_2 use independent draws from the base augmentation pipeline. The shared-base construction used by

SyRC is

$$x_1 = T(\tilde{x}), \qquad x_2 = R_{128}(T(\tilde{x}))\,. \tag{9}$$

Independent pairing removes the shared augmentation draw while preserving the two branch marginals:

$$x_1 = T_1(\tilde{x}), \qquad x_2 = R_{128}(T_2(\tilde{x}))\,, \tag{10}$$

Parallel pairing applies augmentation and resampling separately to the same formed instance:

$$x_1 = T(\tilde{x}), \qquad x_2 = R_{128}(\tilde{x}). \tag{11}$$

Parallel pairing also changes the second branch's marginal by omitting base augmentation. We therefore compare it with and without consistency to measure the loss contribution within that construction.

Table 13 reports these auxiliary designated-run controls. Adding consistency within the parallel construction improves mean target AUC from 92.68% to 93.22%. The shared-base construction reaches 93.61%, but its comparison with parallel pairing is not a matched attribution because the resampled-branch marginal also changes. The class-balanced Δ_R values are source-validation diagnostics computed with the shared DR-128 cache; displacement was not recorded for the independent run.

Table 13: Auxiliary view-construction controls after online formation in designated runs. Avg. is four-target mean video-level AUC (%); class-balanced Δ_R uses the shared DR-128 cache. Checkmarks indicate consistency; dashes denote omitted consistency or unreported displacement.

Pair construction	Consistency	Avg.	Balanced Δ_R
Shared-base, exposure alone	–	92.74	0.03138
Independent: $T_1(\tilde{x})$, $R_{128}(T_2(\tilde{x}))$	✓	93.25	–
Parallel: $T(\tilde{x})$, $R_{128}(\tilde{x})$	–	92.68	0.03231
Parallel: $T(\tilde{x})$, $R_{128}(\tilde{x})$	✓	93.22	0.02843
Shared-base: $T(\tilde{x})$, $R_{128}(T(\tilde{x}))$	✓	**93.61**	**0.02728**

Effect of shared augmentation randomness. We compare pairwise feature discrepancies at the fixed seed-42 online-formation checkpoint using the same FF++ validation cache of 700 videos (140 real and 560 fake). For each of three augmentation-randomness draws, we average frame-level gaps within each video, then within each class, and finally equally across the two class means. The bottom block of Table 5 reports the mean and sample standard deviation across these draws. Writing $z(a)$ for the unit-normalized encoder feature of view a, the cosine gap is

$$g(a, b) = 1 - z(a)^\top z(b). \tag{12}$$

We compute $g_{\text{shared}} = g(T(\tilde{x}), R_{128}(T(\tilde{x})))$ for shared-base pairing and $g_{\text{indep}} = g(T_1(\tilde{x}), R_{128}(T_2(\tilde{x})))$ for independent pairing. As an augmentation-only reference, we also compute $g_{\text{aug}} = g(T_1(\tilde{x}), T_2(\tilde{x}))$ between two independently augmented views.

Shared-base pairing has the smallest cosine gap and independent pairing the largest (Table 5, bottom), with $g_{\text{indep}} > g_{\text{shared}}$ in all three augmentation-randomness draws. At this checkpoint, sharing the base augmentation is associated with a smaller pairwise feature discrepancy.

E Additional Degradation Analysis

We calibrate degradation strengths on 2,099 FF++ validation frames selected using a deterministic first/middle/last-frame rule. The structural similarity index (SSIM) is computed on RGB images before CLIP preprocessing, with data range 255 and an 11×11 Gaussian window of $\sigma = 1.5$. DR-128 uses BOX downsampling followed by bicubic restoration. Gaussian blur uses Pillow radius 1.0, and JPEG uses quality 90 with subsampling 2.

We measure decision-normal displacement using the seed-42 baseline checkpoint on all 22,354 frames from 700 FF++ validation videos. Features are L_2 normalized. Frame-level Δ_G values are first averaged within each video, then within each class, with equal weight assigned to the real and fake class means.

For normalized clean and degraded features z_{clean} and z_G, and the decision normal $\hat{v}$ defined in Section 4.1, the displacement is

$$\Delta_G = \left|\hat{v}^\top (z_G - z_{\text{clean}})\right|. \tag{13}$$

Table 14: Transformation-induced decision-normal displacement at the seed-42 baseline checkpoint. SSIM is measured on the calibration subset; class-balanced Δ_G is measured on the full FF++ validation split.

Transformation	SSIM	Balanced Δ_G
DR-128	0.983638	0.040448
Gaussian blur (radius 1.0)	0.982634	0.045378
JPEG quality 90	0.984014	0.031437

At similar SSIM, Gaussian blur produces larger decision-normal displacement than DR-128, whereas JPEG produces smaller displacement. These measurements describe the checkpoint's immediate response. Appendix F reports the training comparison between DR-128 and Gaussian blur.

F Gaussian-Blur Replacement

To test whether the result depends on the paired transformation, we replace DR-128 with Gaussian blur while retaining the shared-base construction, source mixture, consistency objective, and training budget. Gaussian blur remains part of the common base-augmentation pipeline; this control changes only the transformation used to form the paired view. Table 15 reports the resulting target AUCs.

Table 15: Replacing the paired resampling operation with Gaussian blur in the full pipeline. Values are video-level AUC (%) in designated runs; Avg. is the four-target mean.

Paired transformation	CDF-v1	CDF-v2	DFDCP	DFDC	Avg.
Gaussian blur	95.63	94.07	93.15	85.37	92.06
DR-128	**97.20**	**96.16**	**93.38**	**87.69**	**93.61**

The fixed-checkpoint calibration in Section E further separates image similarity from detector response: at similar structural similarity index (SSIM) values, blur produces larger decision-normal displacement than DR-128. Its lower target AUC after training shows that a larger initial displacement alone does not identify the more effective consistency transformation.

G Additional Resampling Diagnostics

Table 16 reports the values underlying Figure 3(b–c), together with the matched video-score diagnostic. Ranking changes and score differences use DR-128; decision-normal displacement is also evaluated at the neighboring 192, 160, and 96 pixel resolutions.

H Resampling Sensitivity and Score Stability

This appendix derives the relation between the diagnostic in Section 4.1, the consistency loss, and score stability at a fixed checkpoint.

Let w_f, w_r and b_f, b_r be the fake and real class weights and biases of the shared linear head. Define $v = w_f - w_r \neq 0$, $\hat{v} = v/\|v\|_2$, and $\beta = b_f - b_r$. For a unit-normalized encoder feature z, the fake-minus-real logit is $m(z) = v^\top z + \beta$, and the fake probability is $p(z) = \sigma(m(z))$, where $\sigma(t) = (1 + e^{-t})^{-1}$. We assume nonzero encoder outputs so that feature normalization is defined.

Table 16: Source-validation values underlying Figure 3(b–c). Top: class-balanced video-score change and pairwise ranking diagnostics under DR-128. Bottom: class-balanced decision-normal displacement across resampling resolutions.

Diagnostic	Exposure only	SyRC
Mean absolute video-score change	0.0453	0.0382
Adverse reversal (%)	1.383	1.351
Recovered ordering (%)	0.048	0.143
AUC loss (percentage points)	1.335	1.208

Resolution	Exposure only	SyRC
DR-192	0.02117	0.01791
DR-160	0.02470	0.02064
DR-128	0.03138	0.02728
DR-96	0.04539	0.04039

Pairwise displacement and probability change. For normalized base and resampled features (z_b, z_r), let $\ell = (1 - z_b^\top z_r)^2$ be the consistency loss and abbreviate Δ_R as $\Delta = |\hat{v}^\top (z_r - z_b)|$. This absolute projection measures the feature change perpendicular to the real/fake decision boundary. The feature, logit, and probability changes satisfy

$$\Delta \le \|z_r - z_b\|_2 = \sqrt{2(1 - z_b^\top z_r)} = \sqrt{2}\,\ell^{1/4}, \tag{14}$$

$$|m(z_r) - m(z_b)| = \|v\|_2 \Delta, \tag{15}$$

$$|p(z_r) - p(z_b)| \le \frac{\|v\|_2}{4}\Delta \le \frac{\|v\|_2}{2\sqrt{2}}\ell^{1/4}. \tag{16}$$

Proof. The first inequality follows from Cauchy–Schwarz. Since both features have unit norm, $\|z_r - z_b\|_2^2 = 2(1 - z_b^\top z_r)$ and $\ell = \|z_r - z_b\|_2^4/4$, giving the equalities in Equation 14. Subtracting the logits cancels the shared bias β and yields Equation 15. The derivative bound $\sigma'(t) = \sigma(t)(1 - \sigma(t)) \le 1/4$ then gives Equation 16 by the mean value theorem.

Video scores and ranking. For a video V, use the same n_V frames in both conditions and let $S_b(V)$ and $S_r(V)$ denote their mean fake probabilities. For frame i, denote displacement by Δ_i and consistency loss by ℓ_i. Define $\overline{\Delta}_V = n_V^{-1} \sum_i \Delta_i$ and $\varepsilon_V = \|v\|_2 \overline{\Delta}_V/4$. The triangle inequality and Equation 16 give

$$\begin{aligned} |S_r(V) - S_b(V)| \le \varepsilon_V &\le \frac{\|v\|_2}{2\sqrt{2}} \frac{1}{n_V} \sum_{i=1}^{n_V} \ell_i^{1/4} \\ &\le \frac{\|v\|_2}{2\sqrt{2}} \left(\frac{1}{n_V} \sum_{i=1}^{n_V} \ell_i \right)^{1/4} \equiv B_V, \end{aligned} \tag{17}$$

where the last inequality follows from Jensen's inequality for the concave function $t^{1/4}$. The displacement-based quantity ε_V is the tighter intermediate bound, while B_V connects the video-score change directly to the paired consistency loss used for training.

For a fake video V^+ and a real video V^-, define the base score gap as $g_b = S_b(V^+) - S_b(V^-)$. The resampled gap $g_r = S_r(V^+) - S_r(V^-)$ obeys

$$|g_r - g_b| \le \varepsilon_{V^+} + \varepsilon_{V^-} \le B_{V^+} + B_{V^-}. \tag{18}$$

The first bound provides a tighter condition when displacement is measured, whereas the second gives the loss-based condition used in the main text. In particular, a correctly ordered pair retains its ordering whenever

$$g_b > B_{V^+} + B_{V^-}. \tag{19}$$

More generally, $|g_b| > B_{V^+} + B_{V^-}$ preserves the sign of any strict pairwise ordering.

Vulnerable pairs and AUC degradation. The pairwise preservation condition can be applied at each tested resampling strength. Let $S_s(V)$ denote the score of video V after downsampling and restoration at intermediate resolution s, and let $B_s(V)$ denote the loss-based bound in Equation 17 evaluated for the corresponding base–resampled pairs. For a fake video V^+ and a real video V^-, define

$$d_s(V) = S_s(V) - S_b(V), \qquad g_b = S_b(V^+) - S_b(V^-), \qquad g_s = S_s(V^+) - S_s(V^-).$$

The resampled margin satisfies

$$g_s - g_b = d_s(V^+) - d_s(V^-), \qquad |g_s - g_b| \le B_s(V^+) + B_s(V^-). \tag{20}$$

The equality follows directly from the margin definitions. The inequality follows from $|d_s(V)| \le B_s(V)$ and the triangle inequality.

An adverse reversal has $g_b > 0$ and $g_s < 0$. Equation 20 therefore implies $d_s(V^-) - d_s(V^+) > g_b$ and, as a necessary condition,

$$0 < g_b \le B_s(V^+) + B_s(V^-). \tag{21}$$

Hence, every adverse reversal belongs to the set of clean-correct pairs whose margins are no larger than their combined score-change bounds. Membership in this set is not sufficient for reversal, because the two signed score shifts may move the pair in a favorable direction.

Let N_+ and N_- be the numbers of fake and real videos. Define

$$q_{\mathrm{adv}}(s) = \frac{1}{N_+ N_-} \sum_{V^+, V^-} \mathbf{1}\{g_b > 0,\, g_s < 0\}, \tag{22}$$

$$q_{\mathrm{rec}}(s) = \frac{1}{N_+ N_-} \sum_{V^+, V^-} \mathbf{1}\{g_b < 0,\, g_s > 0\}, \tag{23}$$

$$q_{\mathrm{vul}}(s) = \frac{1}{N_+ N_-} \sum_{V^+, V^-} \mathbf{1}\left\{0 < g_b \le B_s(V^+) + B_s(V^-)\right\}. \tag{24}$$

When no pair has an exact score tie, unchanged orderings cancel in the AUC difference, yielding

$$\mathrm{AUC}_b - \mathrm{AUC}_s = q_{\mathrm{adv}}(s) - q_{\mathrm{rec}}(s) \le q_{\mathrm{adv}}(s) \le q_{\mathrm{vul}}(s). \tag{25}$$

The final inequality follows from Equation 21. It separates the observed AUC loss from two intermediate quantities: the adverse reversal rate and the larger set of clean-correct pairs not certified stable by the score bound.

Equation 25 does not imply that AUC must decrease monotonically as resampling becomes stronger. The score-change bounds determine which pairs may reverse, while the signed shifts determine whether they actually do so; recovered orderings further offset adverse reversals in the final AUC. The main text therefore reports both directions of ranking change under DR-128, together with video-score change and decision-normal displacement.