HCSI-Net: Hierarchical Cross-Stream Interaction for Generalizable Deepfake Detection

Marko Brodarič*, Marija Ivanovska*, Deepak Kumar Jain†, Peter Peer*, Vitomir Štruc*
*University of Ljubljana, Ljubljana, Slovenia
†Dalian University of Technology, Dalian, China

***Abstract*—Deepfake detection is increasingly critical for multimedia forensics, yet many detectors degrade under distribution shifts caused by unseen generation pipelines, post-processing, and unconstrained capture conditions. To improve cross-dataset generalization, we propose a powerfull two-stream detector, named HCSI-Net, that couples a CNN and a transformer with progressive interaction during hierarchical feature extraction. The streams are linked via a novel bi-directional spatial cross-gating mechanism that jointly refines local texture cues and global contextual information across stages. The model is trained using manipulation-agnostic supervision based on simulated forgery artifacts and evaluated under challenging cross-dataset evaluation scenarios. Experiments across six widely used datasets demonstrate robust generalization across diverse deepfake generation techniques, achieving a macro-average AUC of 89.13 and consistently outperforming a number of strong state-of-the-art baselines. Ablation results confirm that intermediate cross-stream interaction drives the observed gains.**

***Index Terms*—Deepfake detection, cross-dataset generalization, two-stream networks, CNN–Transformer fusion**

I. Introduction

The rapid progress of generative models has enabled the creation of highly realistic manipulated facial imagery and videos (*deepfakes*) [1]–[3], raising concerns related to fraud, misinformation, and loss of trust in digital media. As a result, deepfake detection has become an increasingly important component in modern biometric and multimedia forensics pipelines. While recent detectors achieve strong performance on the datasets and manipulation types seen during training, they often degrade under distribution shifts caused by unseen generation pipelines, post-processing (e.g., compression and resizing), and in-the-wild capture conditions [4].

A growing body of work therefore focuses on *generalizable* detectors that reduce reliance on generator-specific artifacts. One effective direction is to train with manipulation-agnostic supervision [5], [6], by simulating artifacts, encouraging detectors to learn more robust and transferable cues. However, generalization is influenced not only by data and objectives, but also by architectural bias. Convolutional networks [7] tend to emphasize local, texture-level evidence, whereas vision transformers [8] more naturally integrate global context. Naively fusing these representations only at the output can underutilize their complementarity, since the two branches are not allowed to interact while forming hierarchical features.

In this work, we propose a two-stream deepfake detector that exploits the complementary inductive biases of convolutional and transformer-based vision models through *multi-stage interaction during feature extraction*. The proposed architecture, HCSI-Net, shown in Fig. 1, enables progressive information exchange between local, texture-sensitive features and global contextual representations via a bi-directional spatial cross-gating mechanism, rather than fusing the two streams at the output level. This design allows local and global cues to be jointly refined across the model hierarchy, yielding more discriminative and robust representations under distribution shifts. By explicitly coupling the streams throughout feature learning, the model encourages the emergence of spatially aligned evidence that is less dependent on forgery-specific artifacts. Trained using manipulation-agnostic supervision based on simulated forgery artifacts and evaluated under a strict cross-dataset protocol, HCSI-Net demonstrates strong generalization across public benchmarks. Importantly, the interaction mechanism is architecture-agnostic and compatible with alternative backbones and generalization-oriented training strategies. In summary, we make the following key contributions in this paper:

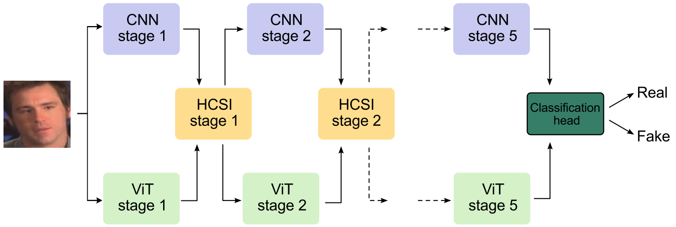

Fig. 1: **HCSI-Net**. An input face crop is processed by CNN and ViT streams, which are coupled at multiple stages via Hierarchical Cross-Stream Interaction (HCSI) to exchange spatial importance and refine complementary local and global cues. The final real/fake decision is obtained by fusing the final representations with a lightweight classification head.

- We develop a two-stream CNN–transformer detector that leverages intermediate cross-stream interactions to jointly refine local and global cues, and is compatible with generalization-oriented training using simulated forgery artifacts as well as evaluation under distribution shifts.
- We introduce a stage-wise, bi-directional spatial cross-gating mechanism for coupling CNN and transformer representations during hierarchical feature learning, enabling explicit spatial interaction across streams.
- We show through extensive experiments and ablations

Supported in parts by ARIS through J2-50065, P2-0250 and P2-0214.

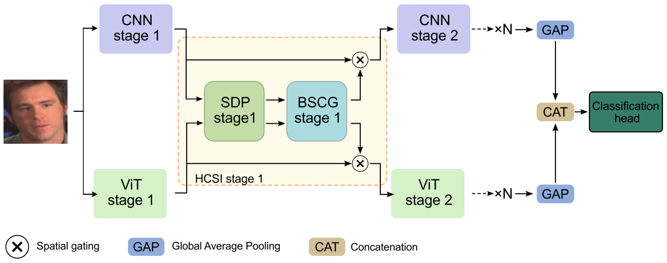

Fig. 2: **Overview of HCSI-Net**. At each stage, CNN and ViT features are projected via shared descriptor projection (SDP) and interacted through bi-directional spatial cross-gating (BSCG) to exchange information. The final prediction of the deepfake detector is obtained by pooling each stream, concatenating features, and applying a classification head.

that the proposed detector leads to consistent improvements over diverse baselines on multiple benchmarks.

II. Related Work

A central challenge in deepfake detection is generalization to unseen manipulations and real-world conditions. While early detectors performed well under in-distribution evaluation, they often relied on shortcut cues that fail to transfer across domains, resulting in degraded performance under distribution shifts. This has motivated work on reducing dependencies on forgery-specific artifacts and improving robustness.

One prominent strategy for improving generalization is to train detectors using synthetic or proxy manipulations that expose intrinsic inconsistencies shared across forgeries, rather than generator-specific traces. Face X-Ray [9] introduced blending-boundary supervision to encourage detectors to focus on common compositing artifacts. Self-Blended Images (SBI) [5] further advanced this idea by generating pseudo-forgeries via self-blending within the same identity, providing broad, manipulation-agnostic supervision. Subsequent work explored softened or bounded discrepancy learning to expose manipulations under distribution shift, such as SeeABLE [6]. These approaches aim to decouple detector training from specific deepfake generation pipelines and have become a common foundation for evaluating cross-dataset performance.

Complementary to data construction, several methods focus on training strategies that explicitly promote generalization. AltFreezing [10] proposed a freezing-based optimization scheme that prevents over-specialization to seen manipulations in video forgery detection. LSDA [11] augmented the latent space to simulate unseen variations, encouraging detectors to transcend forgery specificity. A more radical direction was explored by ProDet [12], which investigated whether robust detectors can be learned while minimizing or even eliminating reliance on deepfake training data altogether, addressing the sustainability challenge posed by rapidly evolving forgery techniques. Self-consistency has also emerged as a recurring theme, with PCL-I2G [13] learning consistency constraints to separate bona fides from deepfakes under shifting conditions.

TABLE I: Stage-wise dimensions for EfficientNet-B4 (CNN stream) and Swin-B (ViT stream) at input 224 × 224.

Stage k	Spatial grid	CNN channels	ViT channels
1	56×56	32	128
2	28×28	56	256
3	14×14	160	512
4	7×7	272	1024
5	7×7	1792	1024

Another line of work improves generalization by learning representations that distill commonalities across diverse forgery processes. UCF [14] explicitly targeted manipulation-agnostic cues by uncovering features shared across different forgery types. CFM [15] argues that detectors should mine critical cues beyond prior forgery knowledge, aiming to discover fundamental indicators that persist across manipulations. FDML [16] addressed entanglement between content, identity, and artifacts through feature disentangling and multi-view learning. Collectively, these methods emphasize that robust detection depends less on memorizing manipulation fingerprints and more on isolating stable inconsistencies that survive changes in generation and post-processing.

In parallel, several approaches designed explicit cues that are difficult for forgeries to preserve, particularly in the temporal or frequency domains. LipForensics [17] focused on mouth dynamics, exploiting the difficulty of synthesizing fine-grained facial motion patterns. NoiseDF [18] modeled noise-related interactions using a multi-head relative-interaction design to detect subtle inconsistencies. More recently, BSF [19] incorporated pixel-wise temporal frequency modeling to capture motion-dependent artifacts beyond spatial spectra. While such cue-driven methods can be highly effective, their robustness often depends on the stability of the assumed cue under compression, resizing, and real-world video pipelines.

Taken together, existing work highlights two recurring gaps: *i)* the need to generalize across unseen manipulations and domains, and *ii)* the need to exploit complementary evidence sources without collapsing into shortcut learning. Many approaches address the former through data design (e.g., SBI [5]), augmentation (e.g., LSDA [11]), or feature mining (e.g., UCF [14], CFM [15]), while others rely on specialized cues (e.g., lip, noise, temporal frequency) [17]–[19].

In contrast to the above surveyed work, our approach directly addresses the limited interaction between local and global representations in existing generalization-oriented detectors. While CNNs and transformers encode complementary local and global cues, combining both models effectively remains challenging. We therefore introduce a two-stream detector with *multi-stage interaction during feature extraction* via bi-directional spatial cross-gating, enabling joint refinement of complementary representations for improved robust cross-domain generalization performance.

III. Methodology

The **proposed HCSI-Net detector** (Fig. 2) builds on the observation that convolutional and transformer-based vision

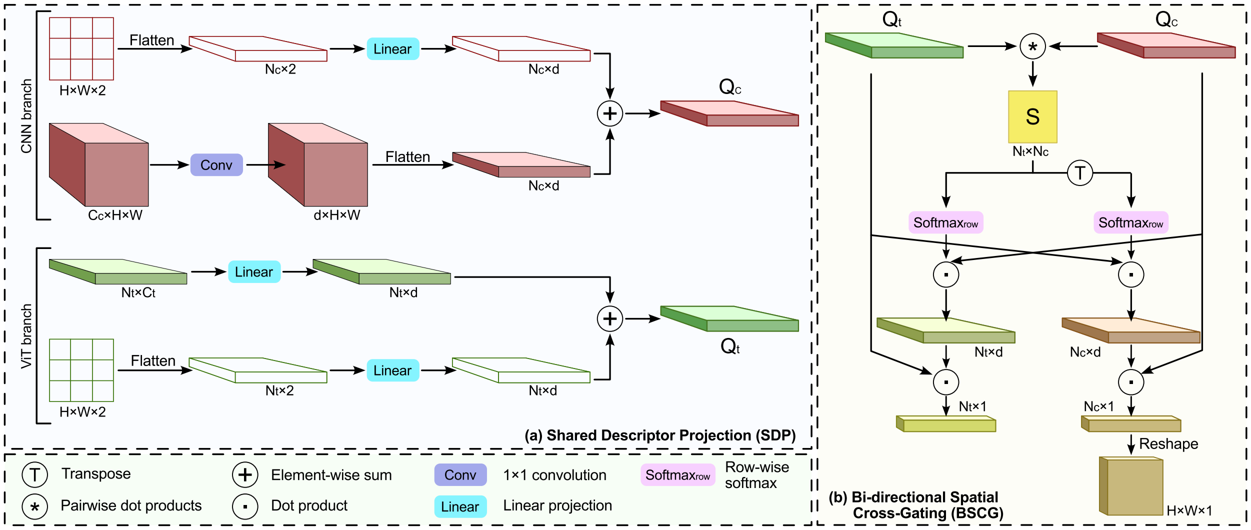

Fig. 3: **Architecture of the HCSI module at each HCSI-Net stage.** (a) Shared descriptor projection (SDP): CNN and ViT features are projected into a shared space for alignment. (b) Bi-directional spatial cross-gating (BSCG): projected features compute cross-stream similarity and generate spatial gates that modulate representations passed to the next stage.

models tend to capture complementary cues. Convolutional neural networks (CNNs) are biased toward local, texture-level evidence through spatially constrained receptive fields, while Visual Transformers (ViTs) integrate information across wider spatial context via token mixing. We leverage this diversity by coupling CNN and ViT streams and allowing them to interact *during* feature extraction, rather than only at the output. Concretely, the proposed model consists of three parts: *i)* a two-stream backbone (EfficientNet-B4 + Swin-B) with access to intermediate stages, *ii)* a stage-wise interaction module composed of *Shared Descriptor Projection (SDP)* and *Bi-directional Spatial Cross-Gating (BSCG)*, and *iii)* a global fusion and prediction head that produces the final decision.

Stage Features and Notation. Let B be the batch size. At stage k (Table I), the CNN stream yields a spatial feature map

$$\mathbf{F}_c^k \in \mathbb{R}^{B\times C_c^k\times H^k\times W^k}, \tag{1}$$

while the ViT stream yields a token sequence arranged on a square grid, as shown in Fig. 3,

$$\mathbf{F}_t^k \in \mathbb{R}^{B\times N^k\times C_t^k}, \qquad N^k = H^k W^k. \tag{2}$$

When needed, we reshape the token sequence into a 2D map $\mathbf{X}_t^k \in \mathbb{R}^{B\times C_t^k\times H^k\times W^k}$ and treat each spatial location (CNN) or token (ViT) as a node. We denote the number of CNN nodes by $N_c^k{=}H^kW^k$ and ViT nodes by $N_t^k{=}H^kW^k$.

Shared Descriptor Projection (SDP). Directly comparing CNN channels and ViT embeddings is not meaningful because they exist in different spaces and have different dimensionalities. We therefore project both streams into a shared descriptor space of dimensionality d (fixed across stages; we use $d{=}64$). The CNN map is projected by a 1×1 convolution and then flattened; the ViT tokens are projected by a linear layer. We ℓ_2-normalize each node descriptor along the descriptor dimension to stabilize dot-product similarities:

$$\begin{aligned}\mathbf{D}_c^k &= \mathrm{norm}\big(\mathrm{flat}\big(\phi_c^k(\mathbf{F}_c^k)\big)\big) \in \mathbb{R}^{B\times N_c^k\times d},\\ \mathbf{D}_t^k &= \mathrm{norm}\big(\phi_t^k(\mathbf{F}_t^k)\big) \in \mathbb{R}^{B\times N_t^k\times d}.\end{aligned} \tag{3}$$

where ϕ_c^k is a 1×1 convolution, ϕ_t^k is a linear projection, $\mathrm{flat}(\cdot)$ maps $B\times d\times H^k\times W^k \to B\times N_c^k\times d$, and $\mathrm{norm}(\cdot)$ denotes ℓ_2 normalization.

Pure content-based matching across all nodes can be ambiguous, especially in early stages with repeated textures. To anchor similarity in spatial layout, we attach a lightweight positional descriptor to each node. We form normalized coordinates on $[-1,1]\times[-1,1]$,

$$\mathbf{p}^k \in \mathbb{R}^{N^k\times 2}, \qquad \mathbf{p}_n^k = (x_n, y_n), \quad x_n, y_n \in [-1,1], \tag{4}$$

project them to the same d-dimensional space using a stage-specific linear projection ψ^k, and ℓ_2-normalize:

$$\begin{aligned}\mathbf{P}_c^k &= \mathrm{norm}\big(\psi_c^k(\mathbf{p}^k)\big) \in \mathbb{R}^{N_c^k\times d},\\ \mathbf{P}_t^k &= \mathrm{norm}\big(\psi_t^k(\mathbf{p}^k)\big) \in \mathbb{R}^{N_t^k\times d}.\end{aligned} \tag{5}$$

We then fuse appearance and position using an element-wise sum with weight β_{pos}:

$$\begin{aligned}\mathbf{Q}_c^k &= \mathrm{norm}\big(\mathbf{D}_c^k + \beta_{\text{pos}}\mathbf{P}_c^k\big) \in \mathbb{R}^{B\times N_c^k\times d},\\ \mathbf{Q}_t^k &= \mathrm{norm}\big(\mathbf{D}_t^k + \beta_{\text{pos}}\mathbf{P}_t^k\big) \in \mathbb{R}^{B\times N_t^k\times d}.\end{aligned} \tag{6}$$

This produces the final node descriptors used by the cross-gating module.

Bi-directional Spatial Cross-Gating (BSCG). Given fused descriptors $\mathbf{Q}_c^k$ and $\mathbf{Q}_t^k$, we compute dense cross-similarities between all CNN nodes and all ViT nodes using scaled dot-products with temperature τ:

$$\mathbf{S}^k = \frac{1}{\tau\sqrt{d}}\,\mathbf{Q}_c^k\,\big(\mathbf{Q}_t^k\big)^\top \in \mathbb{R}^{B\times N_c^k\times N_t^k}, \tag{7}$$

where $(\cdot)^\top$ denotes transpose over the node dimension. We use $\mathbf{S}^k$ for the CNN→ViT direction and its transpose for the opposite direction:

$$\mathbf{S}^k_{c\to t} = \mathbf{S}^k, \qquad \mathbf{S}^k_{t\to c} = \left(\mathbf{S}^k\right)^\top. \tag{8}$$

We obtain attention matrices by applying a row-wise softmax:

$$\begin{aligned} \mathbf{A}^k_{c\to t} &= \mathrm{softmax}_{\mathrm{row}}\big(\mathbf{S}^k_{c\to t}\big) \in \mathbb{R}^{B\times N_c^k\times N_t^k}, \\ \mathbf{A}^k_{t\to c} &= \mathrm{softmax}_{\mathrm{row}}\big(\mathbf{S}^k_{t\to c}\big) \in \mathbb{R}^{B\times N_t^k\times N_c^k}. \end{aligned} \tag{9}$$

We compute cross-attended context vectors via matrix multiplication (cross dot-product):

$$\begin{aligned} \mathbf{V}^k_{c} &= \mathrm{LN}\big(\mathbf{A}^k_{c\to t}\mathbf{Q}^k_t\big) \in \mathbb{R}^{B\times N_c^k\times d}, \\ \mathbf{V}^k_{t} &= \mathrm{LN}\big(\mathbf{A}^k_{t\to c}\mathbf{Q}^k_c\big) \in \mathbb{R}^{B\times N_t^k\times d}. \end{aligned} \tag{10}$$

where $\mathrm{LN}(\cdot)$ is layer normalization, used to stabilize the gating signal across stages.

For each node, we score how consistent the node descriptor is with its retrieved context using a dot-product, then map it to $[0, 1]$ via a sigmoid:

$$\begin{aligned} \mathbf{s}^k_{c} &= \big\langle \mathbf{Q}^k_c,\, \mathbf{V}^k_c \big\rangle \in \mathbb{R}^{B\times N_c^k\times 1}, \\ \mathbf{s}^k_{t} &= \big\langle \mathbf{Q}^k_t,\, \mathbf{V}^k_t \big\rangle \in \mathbb{R}^{B\times N_t^k\times 1}. \end{aligned} \tag{11}$$

$$\mathbf{g}^k_c = \sigma(\mathbf{s}^k_c) \in \mathbb{R}^{B\times N_c^k\times 1},\ \mathbf{g}^k_t = \sigma(\mathbf{s}^k_t) \in \mathbb{R}^{B\times N_t^k\times 1}. \tag{12}$$

We reshape the CNN gate to a spatial mask and broadcast it over channels:

$$\mathbf{G}^k_c = \mathrm{reshape}\big(\mathbf{g}^k_c\big) \in \mathbb{R}^{B\times 1\times H^k\times W^k}, \tag{13}$$

while the ViT gate $\mathbf{g}^k_t$ remains in token form and is broadcast over the token embedding dimension.

Finally, we apply multiplicative gating to the *original* stage outputs and forward the enhanced features to stage $k{+}1$:

$$\begin{aligned} \mathbf{F}^{k\prime}_c &= \mathbf{F}^k_c \odot \mathbf{G}^k_c \in \mathbb{R}^{B\times C_c^k\times H^k\times W^k}, \\ \mathbf{F}^{k\prime}_t &= \mathbf{F}^k_t \odot \mathbf{g}^k_t \in \mathbb{R}^{B\times N_t^k\times C_t^k}. \end{aligned} \tag{14}$$

where $\odot$ denotes element-wise multiplication with broadcasting over the channel/embedding dimension. Repeating SDP+BSCG across $K{=}5$ stages enables multi-scale interaction: early stages exchange fine-grained, local evidence, while later stages exchange more semantic, globally contextual cues.

Global Fusion and Prediction Head. After the final interaction stage ($k = K$), we summarize each stream with global pooling and fuse them into a single representation. The CNN stream is aggregated by global average pooling (GAP) over spatial dimensions, and the ViT stream is aggregated by averaging over tokens:

$$\begin{aligned} \mathbf{z}_c &= \mathrm{GAP}(\mathbf{F}^{K\prime}_c) \in \mathbb{R}^{B\times C_c^K}, \\ \mathbf{z}_t &= \frac{1}{N^K}\sum_{n=1}^{N^K} \mathbf{F}^{K\prime}_t(n) \in \mathbb{R}^{B\times C_t^K}. \end{aligned} \tag{15}$$

We concatenate the pooled descriptors and classify with a compact MLP head:

$$\begin{aligned} \mathbf{z} &= [\mathbf{z}_c; \mathbf{z}_t] \in \mathbb{R}^{B\times (C_c^K + C_t^K)}, \\ \mathbf{o} &= \mathrm{MLP}(\mathbf{z}) \in \mathbb{R}^{B\times 2}. \end{aligned} \tag{16}$$

In our implementation, the MLP is a 3-layer head $2816 \to 1024 \to 512 \to 2$, with BatchNorm, ReLU, and Dropout(0.5) after each hidden layer.

IV. Experimental Setup

Experimental Datasets. To assess generalization, we adopt a strict cross-dataset setting where the model is trained only on SBI pseudo-fakes (generated from FF++ pristine data) and evaluated *without* any fine-tuning on the target benchmark. We report results on the official test splits of the following six widely adopted standard deepfake video datasets: FaceForensics++ (FF++) [21], Celeb-DF v2 (CDF) [1], Deep Fake Detection Dataset (DFD) [2], DeepFake Detection Challenge (DFDC) [22], Deepfake Detection Challenge Preview Dataset (DFDCp) [23], and Face Forensics in the Wild (FFIW) [3].

Performance Scoring. Although our model operates on individual frames, we evaluate the models' performance in a video-level setup. For each video, we sample a fixed number of frames, detect and crop faces, and run the model on the resulting face crops. Frame scores are aggregated into a single video score by averaging across the sampled frames. When multiple faces are present, we use the maximum score within a frame before averaging. This produces one prediction per video, enabling direct computation of video-level ROC statistics.We report the Area Under the ROC Curve (ROC-AUC) computed from the video-level scores for each dataset. To summarize cross-dataset generalization performance with a single scalar value, we also compute the macro-average AUC across all datasets: $\mathrm{mAUC} = \frac{1}{|\mathcal{D}|}\sum_{d\in\mathcal{D}} \mathrm{AUC}(d)$, where $\mathcal{D}$ denotes the set of test datasets considered in the evaluation.

Not all competing methods report video-level AUC on the same set of datasets. To keep comparisons fair, for each baseline method b that reports results on a subset $\mathcal{D}_b \subseteq \mathcal{D}$, we compute our average on the *same* subset:

$$\mathrm{mAUC}^{(\mathrm{ours})}_b = \frac{1}{|\mathcal{D}_b|}\sum_{d\in\mathcal{D}_b} \mathrm{AUC}^{(\mathrm{ours})}(d), \tag{17}$$

and we report the corresponding improvement using $\Delta_b = \mathrm{mAUC}^{(\mathrm{ours})}_b - \mathrm{mAUC}^{(b)}$. This ensures that each reported improvement compares averages over identical datasets, enabling fair and consistent comparison across competing detectors.

Baselines. We compare against a diverse set of detectors that target cross-dataset generalization through data design, training strategies, feature mining, and cue-driven modeling, including AltFreezing [10], FFIW [3], LipForensics [17], LSDA [11], PCL+I2G [13], ProDet [12], UCF [14], CFM [15], NoiseDF [18], FDML [16], BSF [19], LESB [20], SBI [5]. When a method does not report video-level AUC for a particular dataset, we denote the corresponding entry as

TABLE II: **Comparison with the State-Of-The-Art**. Reported are AUC scores (higher is better) for each dataset as well as averages over all available test datasets. Note that HCSI-Net outperforms all competing methods in terms average AUC score.

Method	**Venue**	**FF++**	**CDF**	**DFD**	**DFDC**	**DFDCp**	**FFIW**	**Avg. AUC (↑)**	$\mathbf{mAUC^{(HCSI-Net)}}$ **(↑)**
FFIW [3]	CVPR'21	99.30	78.30	–	–	74.10	–	83.90	**91.54**
LipForensics [17]	CVPR'21	97.10	82.40	–	73.50	–	–	84.33	**88.12**
PCL+I2G [13]	ICCV'21	99.07	90.03	99.07	67.52	74.37	–	86.01	**89.41**
SBI [5]	CVPR'22	99.64	93.18	97.56	72.42	86.15	84.83	88.96	**89.13**
AltFreezing [10]	CVPR'23	98.60	89.50	98.50	–	–	–	95.53	**95.87**
UCF [14]	ICCV'23	–	82.40	94.50	80.50	–	–	85.80	**87.63**
CFM [15]	TIFS'23	–	89.70	95.20	70.60	80.20	83.10	83.76	**87.10**
NoiseDF [18]	AAAI'23	93.99	75.89	–	63.89	–	–	77.92	**88.12**
ProDet [12]	NeurIPS'24	95.91	84.48	–	72.40	81.16	–	83.49	**87.31**
LSDA [11]	CVPR'24	–	83.00	88.00	73.60	81.50	–	81.53	**86.94**
FDML [16]	Neurocomp.'24	99.55	73.10	–	–	–	–	86.33	**94.88**
BSF [19]	ICCV'25	–	89.70	97.30	75.20	–	–	87.40	**87.63**
LESB [20]	WACV'25	–	93.13	–	71.98	–	83.01	82.71	**84.26**
HCSI-Net (Ours)	IWBF'25	99.31	90.45	97.84	74.61	84.85	87.73	–	**89.13**

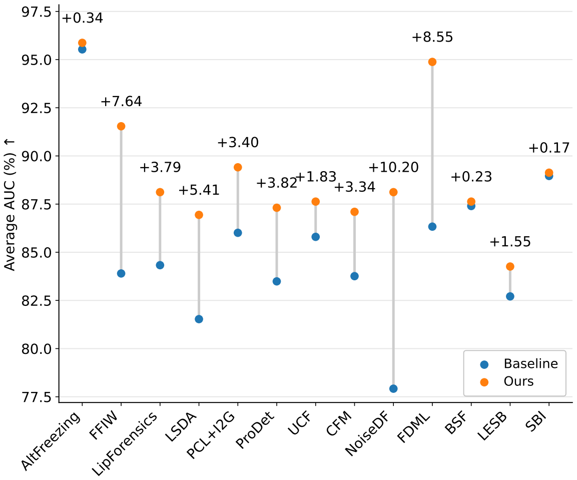

Fig. 4: **Subset-matched performance improvements** over each baseline relative to HCSI-Net, i.e., Δ_b. Positive values indicate performance gains.

unavailable and exclude that dataset from the subset-average ($\mathrm{mAUC}_b^{(\mathrm{ours})}$) computation to ensure a fair comparison.

Training Protocol and Implementation Details. We train our model in two stages using the Self-Blended Images (SBI) protocol [5] and images from FaceForensics++ (FF++) [21]. First, we pretrain the CNN and ViT backbones *independently* on SBI pseudo-forgeries, following the original SBI training recipe. The two backbones are pretrained in separate runs with different random seeds, which yields different effective SBI samples due to the stochastic self-blending process. Second, we initialize the architecture from the pretrained weights and fine-tune it end-to-end on SBI at 224 × 224 resolution with AdamW (learning rate 3×10^{-5}, weight decay 10^{-4}) and a batch size of 16 (paired real/SBI samples per iteration), using a binary classification objective. During fine-tuning, we employ a standard set of strong data augmentations designed to emulate common post-processing distortions (e.g., resampling/blur/noise and photometric perturbations), together with MixUp/CutMix, random erasing, and mild label smoothing. For training stability we use linearly decayed learning-rate schedule. The best model is selected based on the performance on the SBI validation split, monitored in the training phase.

Computational Complexity. At an input resolution of 224 × 224, our two-stream detector comprises 110.9M parameters and requires 17.3G MACs (34.6 GFLOPs) per forward pass. On a single NVIDIA A100-SXM4-80GB GPU, inference with batch size 1 (FP32) takes 34.7 ms on average, with a peak memory footprint of approximately 0.6 GB. With this hardware, training requires around 3.5 minutes per epoch.

V. Results

Quantitative evaluation. Table II summarizes our cross-dataset performance using the macro-average video-level AUC (mAUC) defined in Eq. (2). Across all six evaluation datasets (FF++, CDF, DFD, DFDC, DFDCP, and FFIW), our method achieves an overall mAUC of 89.13. To ensure fair comparison with prior work that reports results on different dataset subsets, we compute our mAUC on the subset used by each baseline, then report the improvement Δ_b.

Overall, our two-stream model with bi-directional spatial cross-gating consistently outperforms a diverse set of competitors spanning data-centric generalization strategies, feature-mining approaches, and cue-driven detectors. The gains are particularly pronounced over methods that rely on more specialized evidence sources, e.g., +10.20 over NoiseDF and +8.55 over FDML in terms of subset-matched mAUC. We also observe strong improvements over representative generalization-oriented baselines such as LSDA (+5.41), ProDet (+3.82), and PCL-I2G (+3.40). Importantly, the improvement is positive even against strong video-focused baselines (AltFreezing: +0.34, BSF: +0.23), indicating that the proposed mechanism provides benefits regardless of the established training protocols and temporal-frequency modeling.

For clarity, Fig. 4 visualizes the subset-matched averages from Table II as a dumbbell plot, highlighting that our improvements are consistent across conceptually different baselines, while varying in magnitude depending on each baseline's underlying assumptions, training strategy, and evidence type.

Ablation study. We further isolate the contribution of each design choice via ablations reported in Table III. Removing inter-stream interactions (i.e., using the same two backbones

TABLE III: **Ablation of branch configurations** and interaction. AUC (%) is the average result across 6 datasets; Δ is the absolute change relative to the full HCSI-Net model.

Setting	AUC	Δ
No interaction	85.78	-3.35
CNN branch	83.51	-5.62
ViT branch	87.93	-1.20
Both branches + interaction	**89.13**	—

but *without* cross-gating and relying on the final fusion only) reduces performance to an mAUC of 85.78, a drop of 3.35 points compared to the full model. This confirms that the gain is not simply due to having two backbones, but is largely driven by enabling feature exchange during learning.

We also evaluate each backbone independently under the same evaluation protocol. Using only the CNN stream yields 83.51 mAUC, while using only the ViT stream yields 87.93 mAUC. While the ViT branch is notably stronger in isolation, the best performance is obtained when *both* branches are coupled through the proposed bi-directional spatial cross-gating, reaching 89.13. These results confirm our observation that CNN and ViT backbones capture complementary cues, showing that explicit, stage-wise interaction is beneficial in cross-dataset evaluation settings, supporting generalization.

Qualitative analysis. Fig. 5 shows representative failure cases in our cross-dataset evaluation setting. We first show two bona fide frames classified as false positives. The first corresponds to a severely low-quality input, where quantization and compression noise introduce artifact-like patterns that resemble common manipulation cues. The second exhibits pronounced motion blur, which disrupts fine texture statistics and leads to an erroneous deepfake prediction. Conversely, we highlight two false negatives where manipulations are highly photorealistic and preserve coherent facial structure and local texture details. These examples indicate that *i)* real-world degradations can mimic synthesis artifacts and *ii)* high-fidelity generations may suppress or mask detectable traces relied upon by current detectors, motivating further robustness to quality variations and stronger cues beyond low-level artifacts.

VI. Conclusion

We presented a novel two-stream deepfake detector, called HCSI-Net, that couples a CNN and a transformer through bi-directional spatial cross-gating to enable intermediate interaction during hierarchical feature learning. The proposed architecture effectively combines local texture cues with global contextual reasoning and demonstrates strong cross-dataset generalization, achieving an overall mAUC of 89.13 across six diverse and challenging benchmarks. Ablation results confirm that the observed gains are driven by explicit stage-wise interaction, while remaining failures are mainly associated with severe quality degradations and photorealistic manipulations.

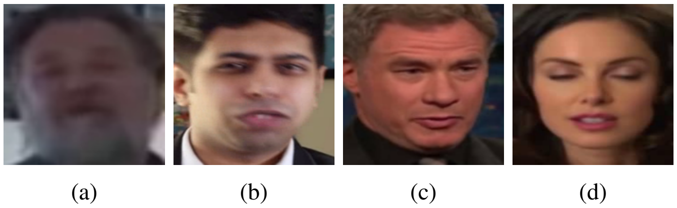

Fig. 5: **Representative failure cases.** (a–b) Bona fide frames predicted as fake (high fakeness score) due to severe quality degradation and blur, introducing artifact-like patterns. (c–d) Deepfake frames predicted as real (low fakeness score), where highly photorealistic manipulations suppress detectable traces.

References

[1] Y. Li, X. Yang, P. Sun, H. Qi, and S. Lyu, "Celeb-df: A large-scale challenging dataset for deepfake forensics," in *IEEE/CVF CVPR*, 2020.

[2] N. Dufour and A. Gully, "Contributing data to deepfake detection research," *Google AI Blog*, 2019.

[3] T. Zhou, W. Wang, Z. Liang, and J. Shen, "Face forensics in the wild," in *IEEE/CVF CVPR*, 2021.

[4] M. Ivanovska and V. Štruc, "On the Vulnerability of Deepfake Detectors to Attacks Generated by Denoising Diffusion Models," in *WACVW 2024*.

[5] K. Shiohara and T. Yamasaki, "Detecting deepfakes with self-blended images," in *IEEE/CVF CVPR*, 2022.

[6] N. Larue, N.-S. Vu, V. Struc, P. Peer, and V. Christophides, "Seeable: Soft discrepancies and bounded contrastive learning for exposing deepfakes," in *IEEE/CVF ICCV*, 2023.

[7] M. Tan and Q. Le, "Efficientnet: Rethinking model scaling for convolutional neural networks," in *ICML*, 2019.

[8] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo, "Swin transformer: Hierarchical vision transformer using shifted windows," in *IEEE/CVF ICCV*, 2021.

[9] L. Li, J. Bao, T. Zhang, H. Yang, D. Chen, F. Wen, and B. Guo, "Face x-ray for more general face forgery detection," in *CVPR*, 2020.

[10] Z. Wang, J. Bao, W. Zhou, W. Wang, and H. Li, "Altfreezing for more general video face forgery detection," in *IEEE/CVF CVPR*, 2023.

[11] Z. Yan, Y. Luo, S. Lyu, Q. Liu, and B. Wu, "Transcending forgery specificity with latent space augmentation for generalizable deepfake detection," in *IEEE/CVF CVPR*, 2024.

[12] J. Cheng, Z. Yan, Y. Zhang, Y. Luo, Z. Wang, and C. Li, "Can we leave deepfake data behind in training deepfake detector?" *NIPS*, 2024.

[13] T. Zhao, X. Xu, M. Xu, H. Ding, Y. Xiong, and W. Xia, "Learning self-consistency for deepfake detection," in *IEEE/CVF ICCV*, 2021.

[14] Z. Yan, Y. Zhang, Y. Fan, and B. Wu, "Ucf: Uncovering common features for generalizable deepfake detection," in *ICCV*, 2023.

[15] A. Luo, C. Kong, J. Huang, Y. Hu, X. Kang, and A. C. Kot, "Beyond the prior forgery knowledge: Mining critical clues for general face forgery detection," *IEEE TIFS*, 2023.

[16] M. Yu, H. Li, J. Yang, X. Li, S. Li, and J. Zhang, "Fdml: Feature disentangling and multi-view learning for face forgery detection," *Neurocomputing*, 2024.

[17] A. Haliassos, K. Vougioukas, S. Petridis, and M. Pantic, "Lips don't lie: A generalisable and robust approach to face forgery detection," in *IEEE/CVF CVPR*, 2021.

[18] T. Wang and K. P. Chow, "Noise based deepfake detection via multi-head relative-interaction," in *AAAI-23*, 2023.

[19] T. Kim, J. Choi, Y. Jeong, H. Noh, J. Yoo, S. Baek, and J. Choi, "Beyond spatial frequency: Pixel-wise temporal frequency-based deepfake video detection," *IEEE/CVF ICCV*, 2025.

[20] E. Soltandoost, R. Plesh, S. Schuckers, P. Peer, and V. Štruc, "Extracting local information from global representations for interpretable deepfake detection," in *IEEE/CVF WACV*, 2025.

[21] A. Rossler, D. Cozzolino, L. Verdoliva, C. Riess, J. Thies, and M. Nießner, "Faceforensics++: Learning to detect manipulated facial images," in *IEEE/CVF ICCV*, 2019.

[22] B. Dolhansky, J. Bitton, B. Pflaum, J. Lu, R. Howes, M. Wang, and C. C. Ferrer, "The deepfake detection challenge (dfdc) dataset," *arXiv preprint:2006.07397*, 2020.

[23] B. Dolhansky, R. Howes, B. Pflaum, N. Baram, and C. C. Ferrer, "The deepfake detection challenge (dfdc) preview dataset," *arXiv preprint:1910.08854*, 2019.