SRRA: Stable-Rank-Based Residual Adaptation for Generalizable Deepfake Detection

Huakun Liu[1], Wenjie Li[1⋆], and Changsheng Xu[2]

[1] School of Computer Science and Engineering, Tianjin University of Technology, Tianjin 300384, China
[2] State Key Laboratory of Multimodal Artificial Intelligence Systems, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China
{huakunliu@stud, wenjieli@email}.tjut.edu.cn, csxu@nlpr.ia.ac.cn

Abstract. Deepfake detection faces challenges in generalization, prompting recent studies to fine-tune pretrained Vision Transformers for improved performance. Existing fine-tuning-based detection approaches often assign the same rank to the tunable residual components of all layers. However, the shallow layers exhibit a distinctly low-rank structure, whereas the middle and deep layers present more complex structures. Therefore, we argue that only slight fine-tuning is required for the shallow layers, which are able to effectively capture low-level forgery cues, while the middle and deep layers require higher residual ranks to model long-range dependencies and extract high-level forgery artifacts. To achieve these goals adaptively, we propose the **S**table-**R**ank-Based **R**esidual **A**daptation (SRRA) strategy. It utilizes singular value decomposition to decompose the pretrained weight matrices and adaptively allocates frozen principal components and learnable residual components based on the **Stable Rank**. In addition, we design the Residual Energy Constraint (REC) and Residual Subspace Orthogonalization (RSO), which suppress excessive perturbations and promote subspace decoupling, avoiding overfitting and improving the generalization performance. Extensive experiments demonstrate that SRRA achieves remarkable improvements in cross-domain generalization across multiple deepfake detection benchmarks. Our code is available at https://github.com/LHK-CodeLab/SRRA.

Keywords: Deepfake detection · Generalization · Fine-tuning · Stable rank · Low-rank adaptation

1 Introduction

In recent years, facial generation and manipulation techniques based on generative adversarial networks and diffusion models have achieved remarkable progress [14, 16, 21], enabling the easy creation and editing of highly realistic facial images and videos. However, the misuse of deepfake technologies has become increasingly prevalent, giving rise to serious security and ethical concerns,

⋆ Corresponding author.

including identity impersonation, opinion manipulation, and evidence fabrication [20, 29, 50]. These threats have eroded public trust in digital media and raised widespread societal concerns. Against this backdrop, reliably and efficiently detecting forged faces has emerged as a challenging research topic in computer vision and multimedia forensics.

Early studies primarily relied on convolutional neural networks (CNNs), with typical architectures such as ResNet, Xception, and EfficientNet [5,15,49] widely adopted for deepfake detection. In addition, many studies integrate auxiliary cues such as frequency, illumination, and identity features [9, 46, 56], and exploit intrinsic artifacts introduced during the generation pipeline [28] to further enhance detection performance. Although these approaches can learn discriminative features and perform well on datasets containing manipulation types similar to those seen during training, their inherently low-rank structures often lead to shortcut learning and overfitting to specific generators or editing routines [52]. When data distributions shift, manipulation techniques evolve, or unseen forgery types emerge, their performance often degrades sharply, revealing limited generalization to diverse forgeries and out-of-domain scenarios.

CNNs possess strong inductive biases toward local patterns, causing them to rely heavily on manipulation-specific edges and textures [13], which ultimately constrains their generalization ability. In contrast, Vision Transformers (ViTs) have weaker inductive biases and can learn more universal semantic relations from large-scale data [10], thus exhibiting stronger cross-domain generalization. Nevertheless, they also require larger datasets for effectively training. Under the limited-data regime typical of deepfake detection, a practical way to mitigate overfitting to a specific manipulation type is to introduce pretrained knowledge and fine-tune the detector to learn forgery-related features. Full-parameter fine-tuning updates all parameters of the detector and offers maximal expressive flexibility. However, given the relatively small size of deepfake datasets, it often leads to forgetting of pretrained priors and overfitting, while incurring substantial computational and storage overhead. To reduce the number of trainable parameters, subsequent works introduced LoRA [18] and its variants [23, 25, 32, 34], which keep the original weight matrix frozen and train only a low-rank matrix with a fixed rank to learn forgery features, combining both to update the model. These approaches significantly reduce the number of trainable parameters and storage requirements while preserving most of the pretrained knowledge. Building on this, Effort [52] further applies Singular Value Decomposition (SVD) to factorize the weight matrix into a frozen principal component and a learnable residual component. It enforces orthogonality and singular value constraints to ensure that residual learning does not distort existing representations, thereby better preserving high-rank semantic structure and stabilizing the feature learning process, leading to superior performance in deepfake detection. However, a major limitation of the above approaches is the **neglect of layer-wise differences in weight distributions** within ViTs. They typically assign the same residual rank to all layers, which limits the detector in adaptively adjusting its representational capacity and feature modeling ability across different depths.

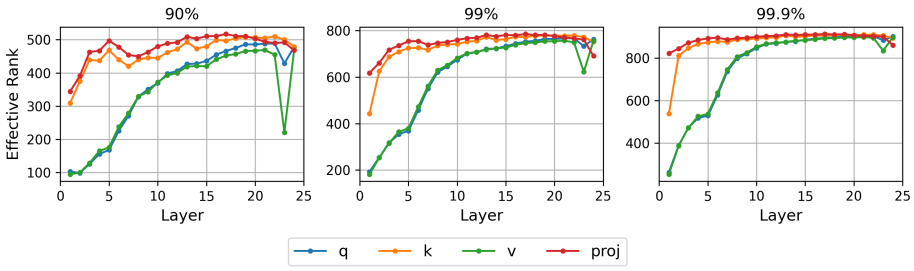

Fig. 1: Illustration of the effective rank across different layers of the pretrained ViT under different matrix energy thresholds (90%, 99%, and 99.9%). The shallow layers show a distinctly low-rank structure, whereas the middle and deep layers require higher ranks to achieve semantic composition.

To analyze the weights across different layers, we perform a singular value spectrum analysis on the attention modules of the ViT in CLIP's visual branch. The effective rank, which quantifies the number of singular values contributing to the overall matrix energy, is employed as a measure. Considering different thresholds for matrix energy, the corresponding results are shown in Fig. 1. As one can see, the shallow layers of the model exhibit a distinctly low-rank structure, indicating that ViTs, after large-scale pretraining, are capable of encoding low-level visual cues such as edges and textures [26,35,41,47]. In contrast, the middle and deep layers require higher ranks to model long-range dependencies and enable semantic composition. Therefore, the fine-tuning process should **adaptively allocate higher ranks to the trainable residual components in middle and deep layers**, enabling the detector to learn more structured high-level artifacts, rather than relearning the low-level features that are already well captured. From the theoretical perspective, Neyshabur et al. [36] proved that different layers contribute unequally to generalization, and the generalization ability is closely related to the norm structure of layer-wise weights, *i.e.*, the stable rank. To this end, we employ the **Stable Rank** [43, 44] for adaptive residual rank allocation. This metric can be used to derive the residual rank through the distribution of singular values, thus avoiding manually setting an energy threshold. The use of stable rank allocates lower residual ranks to shallow layers of deepfake detectors, while higher residual ranks are assigned to middle and deep layers to capture more complex artifacts, leading to improved generalization for deepfake detection.

Building upon the above observations, we propose the **S**table-**R**ank-Based **R**esidual **A**daptation (SRRA) strategy, which consists of Stable Rank Decomposition (SRD) and two constraints on residual energy and residual orthogonality to achieve generalizable deepfake detection. Specifically, in SRD, SVD is first applied to the pretrained weight matrices, which are then decomposed into principal and residual components according to their stable rank. The principal components, which carry the primary discriminative capacity, remain frozen,

while fine-tuning is confined to the residual components. SRD adaptively performs layer-wise residual rank allocation, effectively avoiding the under- and over-parameterization issues and enabling the detector to properly capture previously unseen forgery traces. By using stable rank, SRD allocates larger residual rank to the deeper layers, promoting the learning of forgery-related features. However, it may also introduce irrelevant noise, potentially undermining pretrained representations and discriminative power of the detector. To address this issue, we design the Residual Energy Constraint (REC) and the Residual Subspace Orthogonalization (RSO). REC preserves the total energy while applying nuclear-norm regularization to the singular values of the residual component, thereby suppressing excessive perturbations. RSO enforces orthogonality on the singular vector matrices of the residual component, enhancing subspace decoupling and avoiding collapse onto a few forgery directions. REC and RSO complement each other, preventing overfitting to training-specific forgery artifacts and substantially improving cross-domain generalization.

The main contributions of this paper are as follows:

- Observe that the pretrained ViT exhibits layer-wise differences in weight distribution. The shallow layers display low-rank structures that can effectively capture low-level forgery cues, whereas the middle and deep layers present more complex structures and require higher residual ranks to extract high-level forgery artifacts.
- Propose SRD, which adaptively allocates frozen principal components and learnable residual components based on the stable rank of each weight matrix, ensuring that every layer of the detector attains an appropriate level of representational capacity during fine-tuning.
- Introduce REC and RSO, where REC constrains the energy distribution of weight matrices to suppress excessive adaptations and prevent deviation from the pretrained feature space, while RSO enforces orthogonality on the residual singular vectors to promote subspace decoupling and prevent collapse onto a few forgery directions.

2 Related Work

2.1 Task-Specific Training for Deepfake Detection

Deepfake detection has advanced rapidly in recent years. Early studies primarily focused on architectural modifications to CNNs (Xception, EfficientNet) to guide detectors [1, 37, 42] toward subtle forgery cues such as frequency-domain signatures [11, 27], optical flow [2], and facial geometry [48]. With the advent of large-scale benchmarks like FaceForensics++ (FF++) [42], Celeb-DF (CDF) [29], Deepfake Detection Challenge (DFDC) [7], the field has shifted from case-specific heuristics to systematic evaluation of generalization. Representative methods include: Face X-ray [28], which detects inherent blending boundaries; SRM [33], which uses high-pass filtering to expose high-frequency artifacts; UCF [53], which

disentangles image information and retains shared forgery cues to improve cross-domain robustness; IID [51], which exploits identity-related signals to discriminate forgeries; and LSDA [19], which adopts a teacher–student framework to learn latent forgery representations for better generalization. However, the inherent inductive biases of convolutional architectures often yield low-rank representations that fail to capture diverse forgery patterns under distribution shift, thereby constraining overall generalization.

2.2 Pretrained Fine-Tuning for Deepfake Detection

Concurrently, with the rise of CLIP [40] and other large models, incorporating large-scale pretrained knowledge into detectors via parameter-efficient fine-tuning (PEFT) has become an effective means to mitigate overfitting to a single forgery pattern and to improve cross-domain generalization. The mainstream PEFT paradigms include LoRA [18] and Adapters [17]. LoRA freezes the original weights and overlays low-rank updates rather than directly modifying large matrices, training only a small set of low-rank parameters to markedly reduce cost. Adapters insert compact bottleneck modules into the residual paths of Transformer layers, training only these newly added layers while keeping the backbone intact. Methodologically, MoE-FFD [25] parallelizes LoRA and Adapters within ViT and introduces Mixture-of-Experts (MoE) routing to enhance the separability and representational capacity of forgery cues. OSDFD [24] deploys LoRA in self-attention layers and Adapters in feed-forward layers, leveraging their complementarity to achieve substantial generalization gains with minimal parameter overhead. Forensics Adapter [6] attaches a lightweight side-adapter to CLIP, combining a task-specific blending-boundary objective with tailored interaction strategies to better exploit CLIP. Effort [52] builds on LoRA by performing SVD on detector features, partitioning the representation into a pretrained-knowledge-preserving principal subspace and a forgery-focused residual subspace; with orthogonality and energy constraints, it maintains a higher effective rank, thereby further boosting generalization.

However, prior adaptation schemes typically assign the same rank $r \in \{1, 2, 4 \dots\}$ to all layers, ignoring inter-layer differences in both representational capacity and energy distribution. This often leads to under-allocation of representation capacity in some layers, while others are over-parameterized.

3 Methods

To adaptively allocate appropriate ranks to tunable components during fine-tuning and improve the generalization of the deepfake detection, we propose SRRA. Specifically, during the stage of SRD, pretrained weights are decomposed based on their stable rank, preserving the majority of pretrained knowledge while adaptively allocating residual capacity to meet the fine-tuning requirements of each layer. In addition, we introduce two regularizers: REC, which constrains residual energy, and RSO, which preserves orthogonality among residual singular

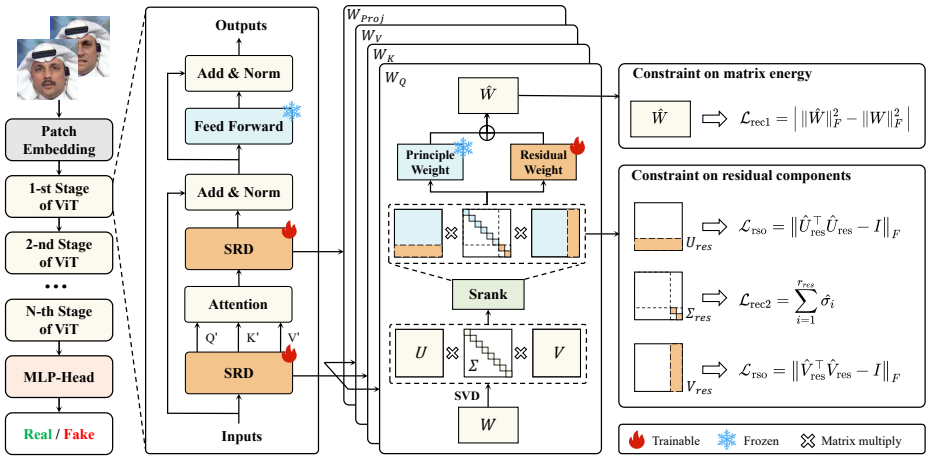

Fig. 2: The overall pipeline of SRRA. SRD decomposes the pretrained weights into a frozen principal component and a trainable residual component based on the stable rank, achieving adaptive residual rank allocation across different layers. REC and RSO are applied to regularize the energy and singular vectors of the residual components, mitigating overfitting and enhancing effective forgery feature extraction.

vectors. These constraints prevent the residual component from distorting the pretrained subspace and encourage the learning of more generalizable forgery representations. The overall pipeline of SRRA is illustrated in Fig. 2.

3.1 Stable Rank Decomposition

Given that the weight matrices of linear layers in ViT's self-attention modules are typically square, we first consider a pretrained weight matrix $W \in \mathbb{R}^{n\times n}$ and factorize it by SVD as

$$W = U\Sigma V^\top, \tag{1}$$

where $U \in \mathbb{R}^{n\times n}$ and $V \in \mathbb{R}^{n\times n}$ are orthogonal matrices whose columns are the left and right singular vectors, satisfying the orthogonality conditions $U^\top U = I_n$ and $V^\top V = I_n$. The matrix $\Sigma \in \mathbb{R}^{n\times n}$ is diagonal with non-negative entries $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_d > 0$, where $d = \text{rank}(W)$ is the matrix rank. SVD provides a convenient basis for separating high-energy and low-energy subspaces, which will be exploited in the following.

On the basis of SVD, we then split the weight matrix W into a principal component W_r and a residual component ΔW based on its singular values, formulated as

$$W = W_r + \Delta W. \tag{2}$$

The principal component $W_r \in \mathbb{R}^{n\times n}$ formed by the top r singular triplets, preserves the dominant representations learned at scale, can be represented as

$$W_r = U_r \Sigma_r V_r^\top, \tag{3}$$

where $U_r \in \mathbb{R}^{n\times r}$, $\Sigma_r \in \mathbb{R}^{r\times r}$, and $V_r \in \mathbb{R}^{n\times r}$. The residual component $\Delta W \in \mathbb{R}^{n\times n}$, constructed from the remaining r_{res} singular triplets, serves as a lightweight adapter to encode forgery-related features, and is defined as

$$\Delta W = U_{res}\Sigma_{res}V_{res}^{\top}, \tag{4}$$

where $U_{res} \in \mathbb{R}^{n\times r_{res}}$, $\Sigma_{res} \in \mathbb{R}^{r_{res}\times r_{res}}$, and $V_{res} \in \mathbb{R}^{n\times r_{res}}$.

To achieve adaptive energy allocation for the principal component and the residual component, we propose to employ the Stable Rank [43] to determine the value of r_{res}. Specifically, the total energy of the weight matrix W is measured by the Frobenius norm $\|W\|_F^2$, and the stable rank is defined as

$$\text{Srank}(W) = \frac{\|W\|_F^2}{\|W\|_2^2} = \frac{\sum_i \sigma_i^2}{\sigma_1^2}, \tag{5}$$

where σ_1 corresponds to the largest singular value. With a hyper-parameter $\tau > 0$, the rank of the residual component is determined by

$$r_{res} \;=\; \lceil \tau \cdot \text{Srank}(W) \rceil\,, \tag{6}$$

where $\lceil\cdot\rceil$ denotes the ceiling operation. It adaptively allocates the residual capacity according to the energy distribution represented by singular values, while preserving the structure of the principal component. Accordingly, the rank of the principal component is obtained by

$$r \;=\; n - r_{res}. \tag{7}$$

During training, we decompose all linear layers within the self-attention modules of ViT, freeze W_r, and optimize only the residual component ΔW. This design preserves the pretrained capacity by keeping the principal subspace intact, while the low-rank, task-focused ΔW learns forgery cues and enables high-level feature composition without distorting the preserved representations.

Based on our analysis in Sec. 1, different layers require varying residual rank levels during the training process. By employing the stable rank defined in Eq. (5), we ensure that each layer attains an appropriate level of expressivity. After applying SRD, smaller residual ranks are assigned to shallow layers that already possess sufficient pretrained representations, and larger ranks are assigned to deeper layers that require high capacity for adaptation.

3.2 Residual Energy Constraint

The application of SRD increases the residual ranks of deeper layers, thereby enhancing the model's expressive capacity. However, when the residual freedom becomes excessively large, the detector tends to overfit to task-irrelevant noise. This issue is particularly evident in deepfake detection scenarios with limited data or imbalanced training samples, thereby weakening the model's generalization across diverse forgery types.

To address this issue, we design the REC with two complementary constraints on the matrix energy. First, we impose a constraint on the total energy of W, defined as

$$\mathcal{L}_{\text{rec1}} = \Big| \|\hat{W}\|_F^2 - \|W\|_F^2 \Big|, \tag{8}$$

where $\hat{W}$ denotes the updated weight matrix, and $|\cdot|$ represents the absolute value operation. This constraint preserves the energy consistency between the updated and pretrained weight matrices, thereby maintaining the representational capacity of the pretrained model. However, this constraint only regularizes the overall energy and does not constrain the singular value spectral structure, which means that dominant singular values may still be weakened and tail singular values amplified during training. As a result, the model may appear numerically stable while actually disrupting the integrity of the pretrained feature space.

As a complementary measure, we then impose a constraint on the energy of the residual component, defined as

$$\mathcal{L}_{\text{rec2}} = \sum_{i=1}^{r_{res}} \hat{\sigma}_i, \tag{9}$$

where $\hat{\sigma}_i$ denotes the updated singular values of the residual weight matrix. The main idea behind the two components of REC is to control the residual energy by regularizing the nuclear norm of the residual component, while preserving the total energy of the pretrained weights. REC effectively suppresses excessive perturbations while enhancing high-level semantic and structural feature composition, thereby enabling more stable and generalizable learning for deepfake detection.

3.3 Residual Subspace Orthogonalization

We adopt a stable-rank–based adaptive decomposition method, which not only enhances the residual rank but also exposes additional trainable singular directions (left/right bases), particularly in the middle and deep layers of the detector. To fully exploit this expanded space for capturing subtle forgery cues while preventing the residual representation from collapsing onto a few forgery directions, we impose an orthogonality constraint on the residual singular bases, formulated as

$$\mathcal{L}_{\text{rso}} = \big\|\hat{U}_{\text{res}}^{\top}\hat{U}_{\text{res}} - I\big\|_F + \big\|\hat{V}_{\text{res}}^{\top}\hat{V}_{\text{res}} - I\big\|_F. \tag{10}$$

It should be noted that RSO has a distinct difference from the orthogonal constraint in previous works [52]. In [52], the orthogonal constraint enforces joint orthogonality between the principal and residual components, which can limit the residual's expressive capacity and bias it toward the subspace of the principal components, potentially hindering forgery feature learning. In contrast, the RSO focuses solely on the residual component, preserving its flexibility to capture various deepfake artifacts, while encouraging orthogonality among residual directions to prevent overfitting to specific forgery cues, thereby improving the generalization capacity.

Moreover, the REC and RSO are tightly coupled. If one naively tunes the singular vectors of the residual component without accounting for the singular values, the residual may overly interfere with the principal component, thereby eroding pretrained knowledge and degrading generalization.

3.4 Loss Function

Overall, the total loss function is defined as the weighted sum of the above loss functions, represented as

$$\mathcal{L} = \mathcal{L}_{\text{cls}} + \frac{1}{m}\sum_{i=1}^{m}\mathcal{L}_{\text{rec1}}^{(i)} + \lambda_1 \frac{1}{m}\sum_{i=1}^{m}\mathcal{L}_{\text{rec2}}^{(i)} + \lambda_2 \frac{1}{m}\sum_{i=1}^{m}\mathcal{L}_{\text{rso}}^{(i)} \tag{11}$$

where $\mathcal{L}_{\text{cls}}$ denotes the cross-entropy loss, λ_1 and λ_2 are hyperparameters that control the contributions of each loss term, and m is the total number of pretrained weight matrices.

4 Experiments

4.1 Settings

Datasets. We evaluate the performance of the proposed method on five widely used deepfake datasets: FaceForensics++ (FF++) [42], CelebDF (CDF) [29], Deepfake Detection Challenge (DFDC) [7], the preview version of DFDC (DFDC-CP) [8], and DeepfakeDetection (DFD). FF++ is a large-scale deepfake dataset that consists of 1,000 pristine videos, from which over 1.8 million forged images generated by multiple manipulation methods are derived. It includes four commonly used forgery techniques: Face2Face (F2F), DeepFakes (DF), FaceSwap (FS), and NeuralTextures (NT). In addition, FF++ provides three levels of compression for these forged images: raw, lightly compressed (c23), and heavily compressed (c40). Following previous studies, we adopt the lightly compressed version (c23) of FF++ for our experiments.

Implementation Details. We adopt CLIP ViT-L/14 as the backbone and initialize it with OpenAI's pretrained weights. Following the DeepfakeBench [54] preprocessing protocol, 32 frames are uniformly sampled and cropped from each video to form the dataset. Input images are resized to 224×224. We train the model using the Adam optimizer with a batch size of 16, a fixed learning rate of 2×10^{-4}, and a weight decay of 5×10^{-4}. Data augmentation comprises horizontal flipping, random rotation, brightness and contrast perturbations, blurring, and JPEG compression. The value of τ in Eq. (6) is fixed to 0.1. All experiments are conducted on an Nvidia GTX 3090 GPU.

Evaluation Metrics. To ensure an objective and fair comparison with state-of-the-art methods, we primarily report the frame-level area under the ROC curve (AUC). In addition, to compare with recent video-based approaches, we also provide the video-level AUC.

Table 1: Cross-dataset comparison on frame-level and video-level (AUC). Best in **bold**, second-best underlined.

Cross-dataset (Frame-Level)							Cross-dataset (Video-Level)						
Method	Venue	CDF	DFDC	DFDCP	DFD	Avg.	Method	Venue	CDF	DFDC	DFDCP	DFD	Avg.
X-ray [28]	CVPR'20	0.679	0.633	0.694	0.766	0.693	F3Net [39]	ECCV'20	0.789	0.718	0.749	0.844	0.775
F3Net [39]	ECCV'20	0.735	0.702	0.735	0.798	0.743	SRM [33]	CVPR'21	0.840	0.695	0.728	0.857	0.787
SPSL [31]	CVPR'21	0.765	0.704	0.741	0.812	0.756	CORE [38]	CVPRW'22	0.809	0.721	0.720	0.882	0.783
SRM [33]	CVPR'21	0.755	0.704	0.741	0.812	0.752	RECCE [4]	CVPR'22	0.823	0.696	0.734	0.891	0.786
RECCE [4]	CVPR'22	0.732	0.713	0.742	0.812	0.750	SBI [45]	CVPR'22	0.932	0.724	0.862	0.976	0.874
UCF [53]	ICCV'23	0.753	0.719	0.759	0.807	0.760	UCF [53]	ICCV'23	0.837	0.742	0.770	0.867	0.804
LoRA [23]	ICCV'23	0.838	0.717	–	0.817	–	IID [19]	CVPR'23	0.838	0.700	0.689	0.939	0.791
ED [3]	AAAI'24	0.864	0.721	0.851	–	–	CDFA [30]	ECCV'24	0.899	0.787	**0.927**	–	–
LSDA [51]	CVPR'24	0.830	0.736	0.815	0.880	0.815	LSDA [51]	CVPR'24	0.903	0.769	0.820	0.936	0.857
UDD [12]	AAAI'25	0.869	0.758	0.856	0.910	0.848	UDD [12]	AAAI'25	0.931	0.812	0.881	0.955	0.895
FreqDebias [22]	CVPR'25	0.836	0.741	0.824	0.868	0.818	StA [55]	CVPR'25	0.947	0.843	0.909	0.965	0.916
Effort [52]	ICML'25	0.901	0.798	–	0.923	–	Effort [52]	ICML'25	0.956	0.843	0.909	0.965	0.918
Ours	–	**0.911**	**0.856**	**0.890**	**0.947**	**0.901**	**Ours**	–	**0.968**	**0.889**	0.923	**0.980**	**0.940**

4.2 Generalization Evaluation

We train the model on FF++ and evaluate it on other deepfake datasets to assess cross-domain generalization, measured by the frame-level AUC. The results are shown in the left part of Tab. 1. To ensure consistent data preprocessing and experimental configurations, we adopt the DeepfakeBench as a standardized evaluation benchmark. The results of LoRA [23], ED [3], LSDA [19], UDD [12], FreqDebias [22], and Effort [52] are taken from their original papers, while the remaining baselines follow the official results reported by DeepfakeBench. Compared with existing methods, SRRA demonstrates superior generalization performance across four unseen datasets, including CDF, DFDC, DFDCP, and DFD. Compared with the state-of-the-art method Effort, SRRA achieves notable performance improvements, showing AUC gains of 1.0%, 5.8%, and 2.4% on CDF, DFDC, and DFD, respectively. It also surpasses the second-best method UDD by 3.4% on DFDCP. Overall, SRRA attains an average AUC improvement of 5.3%, demonstrating its stronger cross-domain generalization capability.

To make the comparison more comprehensive, we additionally report video-level AUC and compare our method with a range of advanced video-based approaches, as shown in the right part of Tab. 1. The results of SBI [45], IID [51], UDD [12], StA [55], and Effort [52] are taken from their original papers, while the remaining results are taken from [52]. As shown in the table, our method consistently achieves superior performance across all four unseen datasets. The largest gain of 4.6% is achieved on DFDC, which is also the largest and most challenging dataset, making the result more convincing. The improvements on CDF, DFDCP, and DFD are 1.2%, 1.4%, and 1.5%, respectively. On average, our method yields a 2.2% improvement across all datasets, clearly demonstrating that SRRA achieves substantial gains in cross-domain generalization for video-level deepfake detection.

Table 2: Ablation studies on the proposed SRD, REC and RSO. Best in **bold**.

SRD	REC	RSO	CDF	DFDC	DFDCP	DFD	Avg.
×	×	×	0.840	0.801	0.800	0.888	0.832
✓	×	×	0.894	0.840	0.849	0.937	0.880
✓	✓	×	0.901	0.843	0.888	0.943	0.894
✓	×	✓	0.889	0.832	0.866	0.936	0.881
✓	✓	✓	**0.911**	**0.856**	**0.890**	**0.947**	**0.901**

Table 3: Ablation studies on different fine-tuning strategies with different residual ranks. Best in **bold**.

Method	r_{res}	CDF	DFDCP	DFD	Avg.
LoRA	1	0.857	0.851	0.942	0.883
	4	0.854	0.865	0.939	0.886
	16	0.865	0.855	0.941	0.887
	64	0.840	0.844	0.934	0.872
Effort	1	0.881	0.865	0.933	0.893
	4	0.892	0.865	0.931	0.896
	16	0.891	0.859	0.941	0.897
	64	0.879	0.856	0.940	0.891
Ours	adaptive	**0.911**	**0.890**	**0.947**	**0.916**

4.3 Ablation Study

Ablation Study on SRD, REC and RSO. To explore the impact of the proposed components within the overall framework, we conduct an ablation study on SRD, REC, and RSO. As shown in Tab. 2, using SRD alone already yields an average AUC that is 4.8% higher than the CLIP baseline. This result demonstrates that the adaptive allocation of principal and residual subspaces in SRD effectively mitigates overfitting. Since both REC and RSO operate on the residual subspace generated by SRD, it is therefore enabled by default in subsequent experiments. When REC is enabled individually, the performance consistently improves on all unseen datasets, increasing the average AUC from 0.880 to 0.894. This confirms that constraining the residual energy prevents excessive deviation from the pretrained subspace. In contrast, enabling RSO only provides limited improvement. Although RSO encourages orthogonality in the residual subspace and promotes the learning of forgery-related variations, it may also amplify residual energy and disturb the principal subspace without the constraint of REC, leading to unstable generalization. When REC and RSO are combined, our method achieves the best overall results, reaching an average AUC of 0.901 and outperforming other configurations on all unseen datasets. This demonstrates the complementary effect of REC and RSO: REC ensures stable optimization by constraining residual energy, while RSO enhances discriminative representation learning. Together, they lead to stronger cross-domain generalization.

Ablation Studies on Different Fine-Tuning Strategies. To validate the effectiveness of the proposed adaptive residual rank allocation mechanism, we compared it with recent fine-tuning approaches. Although Full Fine-Tuning (FFT)

Table 4: Ablation studies on different backbones. Best in **bold**.

Backbone	Method	CDF	DFDC	DFDCP	DFD	Avg.
ViT-B/16	Effort	0.802	0.788	0.811	0.914	0.829
	Ours	**0.846**	**0.798**	**0.829**	**0.921**	**0.849**
ViT-L/14	Effort	0.901	0.798	–	0.923	–
	Ours	**0.911**	**0.856**	**0.890**	**0.947**	**0.901**

provides strong expressive flexibility by updating all model parameters, it introduces substantial computational and memory overhead. When training data are limited or exhibit distribution shifts, FFT is prone to overfitting and may disrupt the general representations learned during pretraining, leading to catastrophic forgetting. As a result, recent studies have increasingly focused on lightweight and stable PEFT approaches, such as LoRA [18] and Effort [52]. Here, we conduct a comparative evaluation against these two approaches, and the corresponding results are presented in Tab. 3.

According to the table, our method achieves the best performance across all unseen datasets (CDF, DFDCP, and DFD) and outperforms the second-best method by an average AUC gain of 1.9%, demonstrating superior cross-dataset generalization capability. LoRA and Effort also demonstrate satisfactory performance, achieving highest AUC scores of 88.7% and 89.7%, respectively, when r_{res} is set to 16. However, these methods typically assign a fixed residual rank to all layers, ignoring the varying representational capacities and adaptation needs across different layers. When the residual rank becomes excessively large, the model can deviate from the pretrained subspace and experience performance degradation. As observed, when $r_{res} = 64$, both LoRA and Effort exhibit decreased generalization performance. Compared with LoRA, Effort explicitly decomposes weight matrices into orthogonal principal and residual subspaces, thus better preserving pretrained knowledge and achieving better generalization performance. Nevertheless, Effort still relies on a globally fixed rank, which limits its adaptability across heterogeneous layers. In contrast, the proposed SRRA adaptively determines the residual rank of each weight matrix based on its energy distribution, enabling effective subspace adaptation while minimizing undesired perturbation to the pretrained weights.

Ablation Studies on Different Backbones. To evaluate the generality and extensibility of our approach, we conducted cross-dataset evaluations using different vision backbones, including CLIP ViT-B/16 (Base) and CLIP ViT-L/14 (Large). For each backbone, two methods were compared, i.e., Effort and SRRA. As shown in Tab. 4, SRRA consistently enhances cross-dataset generalization using different backbones. Specifically, when using the ViT-B/16 as backbone, SRRA achieves the largest improvement of 4.4% on the CDF dataset, with consistent performance gains observed across other datasets as well, highlighting the extensibility of SRRA. Moreover, focusing on SRRA, employing ViT-L/14 as the backbone, which provides a larger model capacity, consistently outperforms ViT-

Table 5: Ablation studies on the hyper-parameters λ_1 and λ_2, measured by AUC. Best in **bold**.

λ_1	λ_2	CDF	DFDC	DFDCP	DFD	Avg.
0.05	0.02	0.896	0.828	0.859	0.941	0.881
0.1	0.05	0.908	0.850	0.890	0.944	0.898
0.1	0.1	0.911	**0.856**	**0.890**	**0.947**	**0.901**
0.1	0.2	**0.912**	0.855	0.888	0.946	0.900
0.2	0.5	0.905	0.834	0.860	0.946	0.886

Table 6: Ablation studies on the hyper-parameter τ, measured by AUC. Best in **bold**.

τ	CDF	DFDC	DFDCP	DFD	Avg.
1	0.875	0.801	0.883	0.917	0.869
1/4	0.902	0.836	**0.896**	0.936	0.893
1/10	**0.911**	**0.856**	0.890	0.947	**0.901**
1/16	0.909	0.853	0.882	**0.948**	0.898

B/16, demonstrating that our method scales effectively with backbone capacity and maintains model-agnostic applicability.

Ablation Studies on Hyper-parameters. We first perform ablation studies on the loss weights λ_1 and λ_2, and the corresponding results are presented in Tab. 5. Under different settings, the cross-dataset performance is measured on four datasets, namely CDF, DFDC, DFDCP, and DFD. Overall, the model maintains relatively stable detection performance across different weight configurations, while moderate settings yield better results, indicating that a proper balance between λ_1 and λ_2 plays an important role in improving cross-dataset generalization. In detail, the best overall performance is achieved when $\lambda_1 = 0.1$ and $\lambda_2 = 0.1$, with an average AUC of 0.901. Accordingly, this configuration is utilized in other experiments. A further observation shows that the performance declines to some extent when the parameter values deviate from this moderate range. For example, when $\lambda_1 = 0.05$ and $\lambda_2 = 0.02$, the average AUC drops to 0.881, while when $\lambda_1 = 0.2$ and $\lambda_2 = 0.5$, the average performance is 0.886, both of which are clearly lower than the optimal setting. In summary, within a reasonable range, the generalization performance is relatively insensitive to these loss weights, which facilitates the applicability of the proposed method across different scenarios.

We further conduct an ablation study on the hyper-parameter τ used to determine the residual rank and evaluate the model performance on the CDF, DFDC, DFDCP, and DFD datasets. As shown in Tab. 6, directly using the stable rank ($\tau = 1$) is not optimal for determining the residual rank, and an appropriate scaling factor is required to achieve the best performance. From these results, we observe that when $\tau = 1/10$, our method attains the best overall performance, yielding a 3.2% improvement compared with the unadjusted setting. Therefore, $\tau = 1/10$ is adopted in all other experiments.

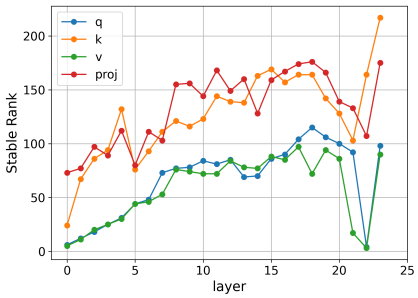

Fig. 3: Illustration of the stable rank used in the proposed SRRA across different layers of ViT.

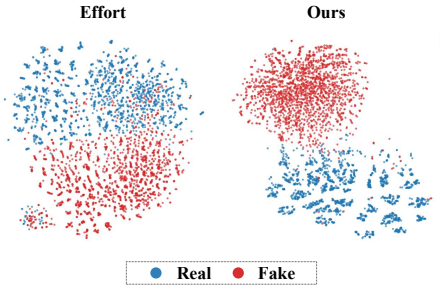

Fig. 4: T-SNE visualization for feature space comparison between Effort and our proposed SRRA.

4.4 Visualizations

Stable Rank Visualizations. We present the distribution of stable ranks across different layers of the proposed deepfake detector trained on the FF++ dataset. As shown in Fig. 3, the stable rank exhibits a hierarchical pattern characterized by low ranks in shallow layers and high ranks in deeper layers, clearly revealing the differences among layer weights. Due to the fact that the stable rank is calculated based on the singular value distribution, similar fluctuations are observed around certain layers, which are consistent with the observations of effective rank shown in Fig. 1. Overall, these results validate that our adaptive residual rank allocation mechanism effectively captures the hierarchical variations, thereby enhancing the model's generalization performance.

t-SNE Visualizations. To intuitively highlight the effectiveness of SRRA, we visualize the t-SNE feature distributions of Effort and our method in Fig. 4, both were trained on the FF++ dataset. Each point represents a sample's feature after t-SNE projection, with blue and red indicating real and fake samples, respectively. As illustrated in the figure, Effort exhibits obvious overlap between real and fake sample features, revealing limited discriminative ability. In contrast, SRRA forms compact and clearly separated clusters, demonstrating its superior capability to capture forgery-related features and achieve better generalization.

5 Conclusion

In this paper, we propose the SRRA framework, achieving superior cross-dataset performance on the deepfake detection task. We first observe that the pretrained ViT exhibits clear layer-wise differences in weight distribution, drawing the conclusion that the shallow layers exhibit low-rank structures capable of capturing low-level forgery cues, while the middle and deep layers possess more complex structures and require higher residual ranks to extract high-level forgery artifacts. Then, to adaptively assign an appropriate rank to the tunable residual component, we design the SRD, which decomposes each weight matrix into a

frozen principal component and a learnable residual component based on the stable rank. To further guide the fine-tuning process for more effective learning of forgery-related features, we introduce REC and RSO. REC preserves the total energy and suppresses excessive perturbations, while RSO promotes subspace decoupling through orthogonality constraints. Extensive experimental results demonstrate that SRRA effectively mitigates overfitting and substantially enhances cross-domain generalization on multiple widely used benchmarks.

Acknowledgements

This work was supported in part by the National Natural Science Foundation of China under Grant 62402339, Grant 62376196, and Grant 62476199; and in part by Tianjin Natural Science Foundation under Grant 24JCJQJC00190.

References

1. Afchar, D., Nozick, V., Yamagishi, J., Echizen, I.: Mesonet: a compact facial video forgery detection network. In: WIFS. pp. 1–7 (2018)
2. Amerini, I., Galteri, L., Caldelli, R., Del Bimbo, A.: Deepfake video detection through optical flow based CNN. In: ICCV (2019)
3. Ba, Z., Liu, Q., Liu, Z., Wu, S., Lin, F., Lu, L., Ren, K.: Exposing the deception: Uncovering more forgery clues for deepfake detection. In: AAAI. vol. 38, pp. 719–728 (2024)
4. Cao, J., Ma, C., Yao, T., Chen, S., Ding, S., Yang, X.: End-to-end reconstruction-classification learning for face forgery detection. In: CVPR. pp. 4113–4122 (2022)
5. Chollet, F.: Xception: Deep learning with depthwise separable convolutions. In: CVPR. pp. 1251–1258 (2017)
6. Cui, X., Li, Y., Luo, A., Zhou, J., Dong, J.: Forensics adapter: Adapting CLIP for generalizable face forgery detection. In: CVPR. pp. 19207–19217 (2025)
7. Dolhansky, B., Bitton, J., Pflaum, B., Lu, J., Howes, R., Wang, M., Ferrer, C.C.: The deepfake detection challenge (DFDC) dataset. arXiv preprint arXiv:2006.07397 (2020)
8. Dolhansky, B., Howes, R., Pflaum, B., Baram, N., Ferrer, C.C.: The deepfake detection challenge (DFDC) preview dataset. arXiv preprint arXiv:1910.08854 (2019)
9. Dong, X., Bao, J., Chen, D., Zhang, T., Zhang, W., Yu, N., Chen, D., Wen, F., Guo, B.: Protecting celebrities from deepfake with identity consistency transformer. In: CVPR. pp. 9468–9478 (2022)
10. Dosovitskiy, A.: An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929 (2020)
11. Frank, J., Eisenhofer, T., Schönherr, L., Fischer, A., Kolossa, D., Holz, T.: Leveraging frequency analysis for deep fake image recognition. In: ICML. pp. 3247–3258 (2020)
12. Fu, X., Yan, Z., Yao, T., Chen, S., Li, X.: Exploring unbiased deepfake detection via token-level shuffling and mixing. In: AAAI. vol. 39, pp. 3040–3048 (2025)
13. Geirhos, R., Rubisch, P., Michaelis, C., Bethge, M., Wichmann, F.A., Brendel, W.: Imagenet-trained CNNs are biased towards texture; increasing shape bias improves accuracy and robustness. In: ICLR (2018)

14. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial networks. Communications of the ACM **63**(11), 139–144 (2020)
15. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: CVPR. pp. 770–778 (2016)
16. Ho, J., Jain, A., Abbeel, P.: Denoising diffusion probabilistic models **33**, 6840–6851 (2020)
17. Houlsby, N., Giurgiu, A., Jastrzebski, S., Morrone, B., De Laroussilhe, Q., Gesmundo, A., Attariyan, M., Gelly, S.: Parameter-efficient transfer learning for nlp. In: ICML. pp. 2790–2799 (2019)
18. Hu, E.J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., Chen, W., et al.: Lora: Low-rank adaptation of large language models. ICLR **1**(2), 3 (2022)
19. Huang, B., Wang, Z., Yang, J., Ai, J., Zou, Q., Wang, Q., Ye, D.: Implicit identity driven deepfake face swapping detection. In: CVPR. pp. 4490–4499 (2023)
20. Jiang, L., Li, R., Wu, W., Qian, C., Loy, C.C.: Deeperforensics-1.0: A large-scale dataset for real-world face forgery detection. In: CVPR. pp. 2889–2898 (2020)
21. Karras, T., Laine, S., Aila, T.: A style-based generator architecture for generative adversarial networks. In: CVPR. pp. 4401–4410 (2019)
22. Kashiani, H., Talemi, N.A., Afghah, F.: Freqdebias: Towards generalizable deepfake detection via consistency-driven frequency debiasing. In: CVPR. pp. 8775–8785 (2025)
23. Kong, C., Li, H., Wang, S.: Enhancing general face forgery detection via vision transformer with low-rank adaptation. In: MIPR. pp. 102–107 (2023)
24. Kong, C., Luo, A., Bao, P., Li, H., Wan, R., Zheng, Z., Rocha, A., Kot, A.C.: Open-set deepfake detection: A parameter-efficient adaptation method with forgery style mixture. arXiv preprint arXiv:2408.12791 (2024)
25. Kong, C., Luo, A., Bao, P., Yu, Y., Li, H., Zheng, Z., Wang, S., Kot, A.C.: Moe-ffd: Mixture of experts for generalized and parameter-efficient face forgery detection. TDSC (2025)
26. Li, A.C., Tian, Y., Chen, B., Pathak, D., Chen, X.: On the surprising effectiveness of attention transfer for vision transformers **37**, 113963–113990 (2024)
27. Li, J., Xie, H., Li, J., Wang, Z., Zhang, Y.: Frequency-aware discriminative feature learning supervised by single-center loss for face forgery detection. In: CVPR. pp. 6458–6467 (2021)
28. Li, L., Bao, J., Zhang, T., Yang, H., Chen, D., Wen, F., Guo, B.: Face x-ray for more general face forgery detection. In: CVPR. pp. 5001–5010 (2020)
29. Li, Y., Yang, X., Sun, P., Qi, H., Lyu, S.: Celeb-df: A large-scale challenging dataset for deepfake forensics. In: CVPR. pp. 3207–3216 (2020)
30. Lin, Y., Song, W., Li, B., Li, Y., Ni, J., Chen, H., Li, Q.: Fake it till you make it: Curricular dynamic forgery augmentations towards general deepfake detection. In: ECCV. pp. 104–122. Springer (2024)
31. Liu, H., Li, X., Zhou, W., Chen, Y., He, Y., Xue, H., Zhang, W., Yu, N.: Spatial-phase shallow learning: Rethinking face forgery detection in frequency domain. In: CVPR. pp. 772–781 (2021)
32. Liu, S.Y., Wang, C.Y., Yin, H., Molchanov, P., Wang, Y.C.F., Cheng, K.T., Chen, M.H.: Dora: Weight-decomposed low-rank adaptation. In: ICML (2024)
33. Luo, Y., Zhang, Y., Yan, J., Liu, W.: Generalizing face forgery detection with high-frequency features. In: CVPR. pp. 16317–16326 (2021)
34. Meng, F., Wang, Z., Zhang, M.: Pissa: Principal singular values and singular vectors adaptation of large language models **37**, 121038–121072 (2024)

35. Merlin, G., Nanda, V., Rawal, R., Toneva, M.: What happens during finetuning of vision transformers: An invariance based investigation. In: CoLLAs. pp. 601–619 (2023)
36. Neyshabur, B., Bhojanapalli, S., Srebro, N.: A pac-bayesian approach to spectrally-normalized margin bounds for neural networks. In: ICLR (2018)
37. Nguyen, H.H., Yamagishi, J., Echizen, I.: Capsule-forensics: Using capsule networks to detect forged images and videos. In: ICASSP. pp. 2307–2311 (2019)
38. Ni, Y., Meng, D., Yu, C., Quan, C., Ren, D., Zhao, Y.: Core: Consistent representation learning for face forgery detection. In: CVPR. pp. 12–21 (2022)
39. Qian, Y., Yin, G., Sheng, L., Chen, Z., Shao, J.: Thinking in frequency: Face forgery detection by mining frequency-aware clues. In: ECCV. pp. 86–103 (2020)
40. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al.: Learning transferable visual models from natural language supervision. In: ICML. pp. 8748–8763 (2021)
41. Raghu, M., Unterthiner, T., Kornblith, S., Zhang, C., Dosovitskiy, A.: Do vision transformers see like convolutional neural networks? **34**, 12116–12128 (2021)
42. Rossler, A., Cozzolino, D., Verdoliva, L., Riess, C., Thies, J., Nießner, M.: Faceforensics++: Learning to detect manipulated facial images. In: ICCV. pp. 1–11 (2019)
43. Rudelson, M., Vershynin, R.: Sampling from large matrices: An approach through geometric functional analysis. JACM **54**(4), 21–es (2007)
44. Shen, S., Zhou, Y., Wei, B., Chang, E.I., Xu, Y., et al.: Tuning stable rank shrinkage: Aiming at the overlooked structural risk in fine-tuning. In: CVPR. pp. 28474–28484 (2024)
45. Shiohara, K., Yamasaki, T.: Detecting deepfakes with self-blended images. In: CVPR. pp. 18720–18729 (2022)
46. Shuai, C., Zhong, J., Wu, S., Lin, F., Wang, Z., Ba, Z., Liu, Z., Cavallaro, L., Ren, K.: Locate and verify: A two-stream network for improved deepfake detection. In: ACM MM. pp. 7131–7142 (2023)
47. Staats, M., Thamm, M., Rosenow, B.: Small singular values matter: A random matrix analysis of transformer models. arXiv preprint arXiv:2410.17770 (2024)
48. Sun, Z., Han, Y., Hua, Z., Ruan, N., Jia, W.: Improving the efficiency and robustness of deepfakes detection through precise geometric features. In: CVPR. pp. 3609–3618 (2021)
49. Tan, M., Le, Q.: Efficientnet: Rethinking model scaling for convolutional neural networks. In: ICML. pp. 6105–6114 (2019)
50. Thies, J., Zollhöfer, M., Nießner, M.: Deferred neural rendering: Image synthesis using neural textures. ACM TOG **38**(4), 1–12 (2019)
51. Yan, Z., Luo, Y., Lyu, S., Liu, Q., Wu, B.: Transcending forgery specificity with latent space augmentation for generalizable deepfake detection. In: CVPR. pp. 8984–8994 (2024)
52. Yan, Z., Wang, J., Jin, P., Zhang, K.Y., Liu, C., Chen, S., Yao, T., Ding, S., Wu, B., Yuan, L.: Orthogonal subspace decomposition for generalizable ai-generated image detection. In: ICML. pp. 70268–70288. PMLR (2025)
53. Yan, Z., Zhang, Y., Fan, Y., Wu, B.: Ucf: Uncovering common features for generalizable deepfake detection. In: ICCV. pp. 22412–22423 (2023)
54. Yan, Z., Zhang, Y., Yuan, X., Lyu, S., Wu, B.: Deepfakebench: A comprehensive benchmark of deepfake detection. arXiv preprint arXiv:2307.01426 (2023)
55. Yan, Z., Zhao, Y., Chen, S., Guo, M., Fu, X., Yao, T., Ding, S., Wu, Y., Yuan, L.: Generalizing deepfake video detection with plug-and-play: Video-level blending and spatiotemporal adapter tuning. In: CVPR. pp. 12615–12625 (2025)

56. Zhu, C., Zhang, B., Yin, Q., Yin, C., Lu, W.: Deepfake detection via inter-frame inconsistency recomposition and enhancement. Pattern Recognition **147**, 110077 (2024)