Exploring Disentangled Content Information for Face Forgery Detection

Jiahao Liang[1], Huafeng Shi[2], and Weihong Deng[1*]

[1] Beijing University of Posts and Telecommunications
[2] SenseTime Research
{jiahao.liang, whdeng}@bupt.edu.cn, shihuafeng1@sensetime.com

Abstract. Convolutional neural network based face forgery detection methods have achieved remarkable results during training, but struggled to maintain comparable performance during testing. We observe that the detector is prone to focus more on content information than artifact traces, suggesting that the detector is sensitive to the intrinsic bias of the dataset, which leads to severe overfitting. Motivated by this key observation, we design an easily embeddable disentanglement framework for content information removal, and further propose a *Content Consistency Constraint* (C^2C) and a *Global Representation Contrastive Constraint* (GRCC) to enhance the independence of disentangled features. Furthermore, we cleverly construct two unbalanced datasets to investigate the impact of the content bias. Extensive visualizations and experiments demonstrate that our framework can not only ignore the interference of content information, but also guide the detector to mine suspicious artifact traces and achieve competitive performance.

Keywords: Face Forgery Detection, Content Information, Disentangled Representation

1 Introduction

With the incredible success of deep learning, numerous techniques for forgery have emerged, such as Deepfakes[14], Face2Face [42], and FaceSwap [30]. Due to the extremely low barriers and easy accessibility, generative techniques are gradually being misused [1, 2].

To defend against, face forgery detection has attracted increasing attention. Early works [22, 47, 32, 10, 17] used hand-crafted facial features (*e.g.*, eyes blinking, head poses, lip movements, *etc.*) to capture some visual artifacts and inconsistencies resulting from the forgery generation process. Meanwhile, some works [11, 35, 24] explored PPG signals representing heart rate information. Later, learning-based methods [48, 45, 13, 7, 46] have made significant progress. Nevertheless, these methods are vulnerable to image compression or noise interference. Frank *et al.* [15] found that, compared to the time domain, mining forgery information in the frequency domain can still maintain satisfactory results even under severe compression [15, 36, 31, 25, 20].

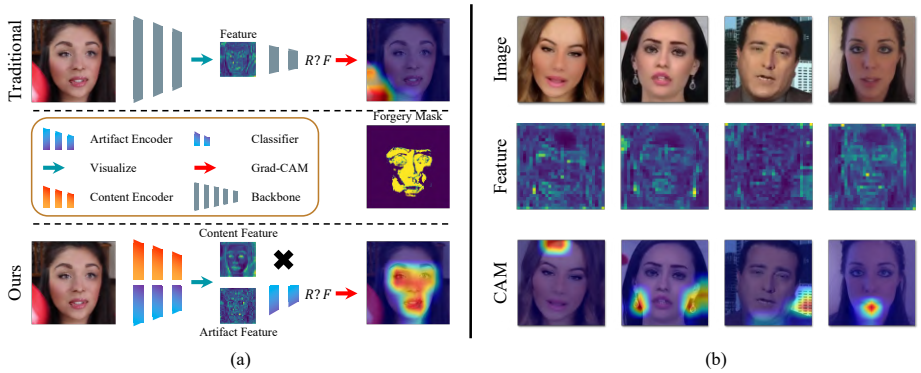


Fig. 1. (a) Unlike the traditional methods (upper), we propose a disentanglement framework (lower) for content information removal. Grad-CAM [39] shows that the traditional detector is distracted by the red object, while our method still mine suspicious artifact traces and the activation region almost consistent with the mask. (b) Visualization of the image (first row), traditional detector's (Xception) features (second row) and Grad-CAM [39] (third row).

We observe that most methods [53, 36, 31, 35] perform admirably in in-dataset evaluations but struggle to maintain comparable results in cross-domain evaluations, which inspire us to conduct an in-depth analysis of the previous method. Existing methods take it for granted that after proper training, the detector will selectively grasp artifact traces as the basis for authenticity judgment. However, the visualization (shown in Figure 1 (b)) illustrates that the feature of the detector remains recognizable content clues, and the detector is prone to overfitting to small local regions, or even focusing only on content information outside the face region.

Based on this key observation, we conjecture that detectors may no longer mine hard-to-capture artifact traces, and instead overfit certain non-artifact (*i.e.*, content) information, thus leading to the failure of cross-domain evaluations.

Therefore, we propose an easily embeddable framework for disentangling content features and artifact features, and only the disentangled artifact features for face forgery detection, thus ignoring the interference of content information. A brief comparison between the traditional methods and our framework is sketched in Figure 1 (a).

However, most disentanglement methods [51, 26, 33] consider only the completeness of features, but do not explore the independence of disentangled features in-depth, which leads to the failure of the face forgery detection (see Table 4). To enhance it, we propose a *Content Consistency Constraint* (C^2C) to ensure that the disentangled features contain the corresponding information and a *Global Representation Contrastive Constraint* (GRCC) to further ensure the purity of the disentangled features, which helps our disentanglement framework to achieve competitive performance. Furthermore, we cleverly construct two un-

balanced datasets based on the FaceForensics++ [38] to investigate the impact of content bias, and further demonstrate that our framework can ignore the interference of the content bias. Notably, our framework is easily embeddable, we embed some backbones into our framework for extensive evaluations and ablation experiments, and experimental results demonstrate the effectiveness and generalization capability of our framework in face forgery detection.

The contributions of this paper could be summarized as three-fold:

- To the best of our knowledge, we are the first to explore the impact of content information on the generalization performance of face forgery detection, and cleverly construct two unbalanced datasets to further investigate the impact of content bias, which brings a novel perspective for this field.
- We design an easily embeddable disentanglement framework for content information removal, and further propose a *Content Consistency Constraint* (C^2C) and *Global Representation Contrastive Constraint* (GRCC) to enhance the independence of disentangled features.
- Extensive visualizations and experiments demonstrate that our framework can not only ignore the interference of content information, but also guide the detector to mine suspicious artifact traces and achieve competitive performance in face forgery detection.

2 Related Works

2.1 Forgery Detection

Benefiting from the great progress of GAN, forgery techniques, especially for faces, have been incredibly advanced. To avoid its illegal use, researchers have explored forgery detection extensively [21, 54, 28, 40, 4].

Later, various learning-based methods [48, 45, 13, 7, 46] demonstrated significant improvements. In addition, some works [3, 27, 8] suggested that shallow local texture details and correlations between local regions of the face can better reflect forgery information. However, almost all of these CNN-based methods only utilize spatial domain information (*i.e.*, RGB, YUV, HSV), and therefore the performance is sensitive to the quality and distribution of the dataset. To counter it, some works [15, 36, 31, 25, 20] transformed images into the frequency domain by DCT transform and analyzed the frequency domain statistics, achieving satisfactory results even with severe compression. Recent attempts to boost the generalization of face forgery detection by extending the activated attention region of the network. Zhao *et al.* [53] proposed multiple spatial attention heads to guide the network focus on different local regions. Wang *et al.* [44] encouraged detectors to dig deeper into previously overlooked regions by masking the sensitive facial regions. Although these CNN-based methods significantly enhance the feature extraction capability of the detector, due to the neglect of the content bias implied in the features, the detector is trapped in the intrinsic bias of the dataset, thus hindering the improvement of cross-domain generalization performance.

2.2 Disentangled Representation

Disentangled representation learning is to decompose complex dimensional coupled information into simple features with a strong distinguishing ability [6]. DR-GAN [43] disentangled the face into identity and pose features for synthesizing faces in arbitrary poses to aid in recognition. Niu *et al.* [33] proposed a cross-verified feature disentangling strategy with robust multi-task physiological measurements. Zhang *et al.* [52] also adopted a similar structure to disentangle pose and appearance features from gait videos. In the field of face anti-spoofing, Zhang *et al.* [51] decompose the facial image into content features and liveness features and introduced LBP map, depth map as auxiliary supervision. Liu *et al.* [26] proposed a new adversarial learning framework to separate the spoof trace into a hierarchical combination of multi-scale patterns. In this paper, we further propose a *Content Consistency Constraint* (C^2C) and *Global Representation Contrastive Constraint* (GRCC) to enhance the independence of disentangled features. And the disentanglement framework only serves as an underlying architecture, a detailed ablation analysis can be found in Section 4.3.

3 Methods

3.1 Motivation

Consider a forged image, which consists of artifact traces and content information, where the content information can be subdivided into identity information and background information. The only difference between the forgery image and the real image is the presence of artifact traces, which is the basis for the detector to determine the authenticity.

We observe that most detectors perform admirably in in-dataset evaluations but struggle to maintain comparable results in cross-dataset evaluation. For further exploration, we visualize the features of the middle layer of the detector and the Grad-CAM [39]. The visualization results (see Figure 1 (b)) illustrate that the feature of the detector remains recognizable content information, and the detector is prone to overfitting to small local regions (DF, NT), or even focusing on content information outside the face region (FS).

Based on this key observation, we conjecture that with the weak constraint of binary labels alone, detectors may no longer mine hard-to-capture artifact traces, and instead overfit certain non-artifact information (*i.e.*, content information), thus failing in cross-dataset evaluations.

Therefore, we propose an embeddable disentanglement framework that disentangles content and artifact features, and the artifact features are used for forgery detection, thus eliminating the interference of content information.

3.2 Basic Disentanglement Framework

We assume that the high-dimensional latent representation of an image consists of content and artifact features. The main purpose is to disentangle them and use the disentangled artifact features for subsequent detection.

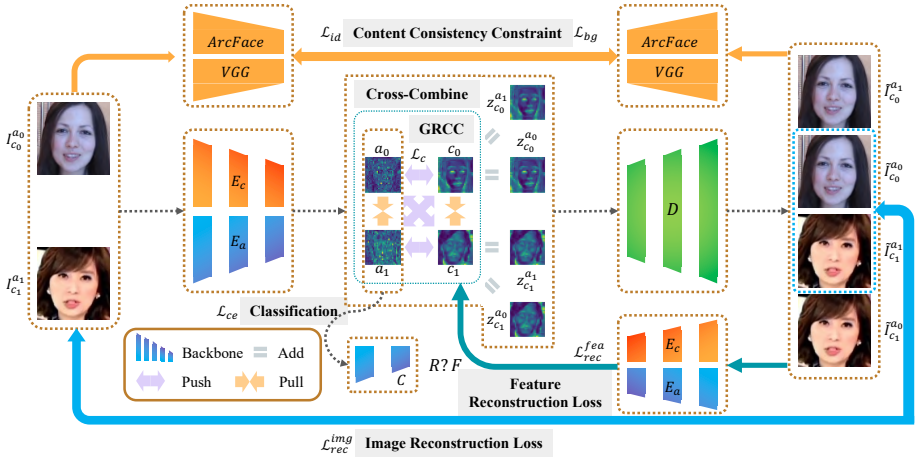


Fig. 2. The overview framework of our method. The input of our network is a pair of images. First, we use artifact encoder E_a and the content encoder E_c to disentangle the content and artifact features, respectively. Then, we feed the artifact features a_0 and a_1 to the classifier C for detection to compute $\mathcal{L}_{ce}$. Next, *Global Representation Contrastive Constraint* (GRCC) is used to compute $\mathcal{L}_c$, and the artifact and content features are cross-combined to get latent representation z_c^a and then reconstruct the images. Finally, the reconstruction loss $\mathcal{L}_{rec}^{img}$, $\mathcal{L}_{rec}^{fea}$ and the *Content Consistency Constraint* (C^2C) loss $\mathcal{L}_{id}$, $\mathcal{L}_{env}$ are calculated to ensure the completeness and independence of the disentangled features.

The disentanglement framework mainly consists of two independent encoders E_c and E_a, for extracting content and artifact features, respectively, a decoder D for the reconstruction of the images, and a classifier C for face forgery detection. Among them, we use the front and back parts of the backbone as the artifact encoder E_a and the classifier C, and the artifact encoder E_a and the content encoder E_c have the same structure, but the parameters are not shared.

Specifically, as shown in Figure 2, with pairwise input images $I_{c_0}^{a_0}$, $I_{c_1}^{a_1}$, where a_0, a_1 and c_0, c_1 denotes the corresponding artifact and content features of the image, respectively. It is worth noting that one of the images is real and the other one is fake. We first use the content encoder E_c and the artifact encoder E_a to get the content features c_0, c_1 and the artifact features a_0, a_1, and the formula is as follow:

$$c_i = E_c(I_{c_i}^{a_i}), a_i = E_a(I_{c_i}^{a_i}), \tag{1}$$

where i denotes the index of feature.

Self-Reconstruction. Then element-wise addition is applied to the content and artifact features encoded from the same image to obtain the high-dimensional latent representation features of the image, *i.e.*, $z_{c_i}^{a_i} = a_i + c_i$. Next, $z_{c_i}^{a_i}$ is fed into the decoder D to reconstruct the corresponding original image $\widetilde{I}_{c_i}^{a_i}$, and the

formula is as follow:

$$\widetilde{I}_{c_i}^{a_i} = D(z_{c_i}^{a_i}). \tag{2}$$

Cross-Reconstruction. Moreover, we **cross-combine** content and artifact features from different images to obtain the high-dimensional latent representation features, *i.e.*, $z_{c_{1-i}}^{a_i} = a_i + c_{1-i}$. Also, $z_{c_{1-i}}^{a_i}$ is fed into the decoder D to reconstruct the image $\widetilde{I}_{c_{1-i}}^{a_i}$, and the formula is as follow:

$$\widetilde{I}_{c_{1-i}}^{a_i} = D(z_{c_{1-i}}^{a_i}). \tag{3}$$

Reconstruction Loss. The decoder D should effectively reconstruct the original image to ensure the completeness of the high-dimensional latent representation feature, so the image reconstruction loss is formulated as:

$$\mathcal{L}_{rec}^{img} = \sum_{i=0}^{1} \left\| I_{c_i}^{a_i} - \widetilde{I}_{c_i}^{a_i} \right\|_1. \tag{4}$$

Image reconstruction loss ensures that the reconstructed image and the original image are consistent at the pixel level. In addition, the encoded features of the reconstructed image should still be consistent with the reconstructed features, so we introduce a feature reconstruction loss:

$$\begin{aligned} \mathcal{L}_{rec}^{fea} = \sum_{i=0}^{1} (& \left\| E_c(\widetilde{I}_{c_i}^{a_i}) - c_i \right\|_1 + \left\| E_a(\widetilde{I}_{c_i}^{a_i}) - a_i) \right\|_1 \\ & + \left\| E_c(\widetilde{I}_{c_{1-i}}^{a_i}) - c_{1-i} \right\|_1 + \left\| E_a(\widetilde{I}_{c_{1-i}}^{a_i}) - a_i) \right\|_1). \end{aligned} \tag{5}$$

3.3 Enhanced Independence of Disentangled Features

Although reconstruction loss can guarantee the completeness of features for the combination of content and artifact features. However, there are still two elements that cannot be guaranteed: (i) Whether the encoders can selectively disentangle features (*i.e.*, whether the disentangled features contain the corresponding information). (ii) Whether the disentangled features contain **only** the corresponding information. We are keenly aware that the key to successful disentangling lies in the establishment of these two conditions, which is proved by subsequent ablation study (Section 4.3). Unfortunately, none of the previous related methods [51, 26, 33] have explored the independence of features in depth. We propose a *Content Consistency Constraint* (C^2C) and a *Global Representation Contrastive Constraint* (GRCC) to further enhance the independence of disentangled features.

Content Consistency Constraint. In cross-reconstruction, content features should determine the background and face ID information of the reconstructed image. Specifically, the cross-reconstructed image $\widetilde{I}_{c_i}^{a_{1-i}}$ should have the same content attributes as the origin image $I_{c_i}^{a_i}$ that encodes the content features c_i.

As we mentioned before, content features consist of background and face ID, so the *Content Consistency Constraint* (C^2C) can be formulated as:

$$\begin{aligned} \text{Content}(\widetilde{I}_{c_i}^{a_{1-i}}) &= \text{Content}(I_{c_i}^{a_i}), \\ \Updownarrow & \\ \text{Identity}(\widetilde{I}_{c_i}^{a_{1-i}}) &= \text{Identity}(I_{c_i}^{a_i}), \\ \text{Backgroud}(\widetilde{I}_{c_i}^{a_{1-i}}) &= \text{Backgroud}(I_{c_i}^{a_i}), \end{aligned} \tag{6}$$

based on this prior condition, we adopt the identity preservation loss $\mathcal{L}_{id}$ and the content perception loss $\mathcal{L}_{bg}$ to preserve the content attributes of the cross-reconstructed images. It is formulated as:

$$\begin{aligned} \mathcal{L}_{id} &= 1 - \cos(\text{ArcFace}(\widetilde{I}_{c_i}^{a_{1-i}}), \text{ArcFace}(I_{c_i}^{a_i})), \\ \mathcal{L}_{bg} &= \left\| \text{VGG}(\widetilde{I}_{c_i}^{a_{1-i}}) - \text{VGG}(I_{c_i}^{a_i}) \right\|_1, \end{aligned} \tag{7}$$

where $\text{ArcFace}(\cdot)$ and $\text{VGG}(\cdot)$ represents a pretrained VGG network and a pretrained ArcFace network, respectively, $\cos(\cdot,\cdot)$ represents the cosine similarity of two vectors. Here $\text{VGG}(\cdot)$ is considered to extract high-level semantic features, and since artifacts are mainly concentrated in low-level texture details [27, 53], the extracted content features is pure and does not contain artifact information.

Global Representation Contrastive Constraint. Artifact features and content features should be two fundamentally distinct spaces. In other words, artifact features and content features can be regarded as two different classes, and the inter-classes feature distance should be much larger than the intra-class feature distance. Specifically, we regard the intra-class features as positive pairs and inter-class features as negative pairs, and adopt the contrastive learning protocol to further eliminate the possible overlap of content features and artifact features. Inspired by [27], we take the Gram matrix of content and artifact features as a global and distinctive representation:

$$\mathbf{G} = (F_i^T F_j)_{n\times n} = \begin{bmatrix} F_1^T F_1 & \cdots & F_1^T F_n \\ \vdots & \ddots & \vdots \\ F_n^T F_1 & \cdots & F_n^T F_n \end{bmatrix}, \tag{8}$$

where F denotes the feature, and n denotes the channel of the feature. For feature distance measurement, we adopt the cosine distance, where closer features render larger scores. Finally, we take the advantage of the InfoNCE [34] to construct a *Global Representation Contrastive Constraint* (GRCC) between the artifact and content features:

$$\begin{aligned} \mathcal{L}_c = &- \log[\frac{\exp(d(\mathbf{G}_{a_0}, \mathbf{G}_{a_1}))}{\exp(d(\mathbf{G}_{a_0}, \mathbf{G}_{a_1})) + \sum_{i=0}^{1} \exp(d(\mathbf{G}_{a_i}, \mathbf{G}_{c_{1-i}}))}] \\ &- \log[\frac{\exp(d(\mathbf{G}_{c_0}, \mathbf{G}_{c_1}))}{\exp(d(\mathbf{G}_{c_0}, \mathbf{G}_{c_1})) + \sum_{i=0}^{1} \exp(d(\mathbf{G}_{a_i}, \mathbf{G}_{c_{1-i}}))}], \end{aligned} \tag{9}$$

where $\mathbf{G}_{a_i}$ and $\mathbf{G}_{c_i}$ represent the flattened vector of the gram matrix of a_i and c_i, respectively, and $d(\cdot,\cdot)$ represents the cosine similarity.

3.4 Overall Loss

The final loss function of the training process is the weighted sum of the above loss functions.

$$\mathcal{L} = \mathcal{L}_{ce} + \lambda_1 \mathcal{L}_{rec}^{img} + \lambda_2 \mathcal{L}_{rec}^{fea} + \lambda_3 \mathcal{L}_{id} + \lambda_4 \mathcal{L}_{bg} + \lambda_5 \mathcal{L}_c, \tag{10}$$

where $\mathcal{L}_{ce}$ denotes the cross entropy loss, λ_1, λ_2, λ_3, λ_4, λ_5 are the weights for balancing the loss.

4 Experiments

4.1 Experimental Setting

Datasets. To validate the effectiveness of our method, we choose the most widely used benchmark FaceForensics++ (FF++) [38] for training. It contains 1 real sub-dataset and 4 fake sub-datasets, *i.e.*, Deepfakes (DF) [14], Face2Face (FF) [42], FaceSwap (FS) [30] and NeuralTextures (NT) [41]. Each sub-dataset contains 1,000 videos, and we follow the official standard by using 720 videos for training, 140 videos for validation, and 140 videos for testing, and we adopt the LQ version by default and specify the version otherwise. Celeb-DF [23] uses 59 celebrity interview videos on YouTube as the original videos. In total, 590 real videos and 5,639 DeepFakes videos are included.
Metrics. We apply the accuracy score (ACC), equal error rate (EER), and the area under the receiver operating characteristic (ROC) Curve (AUC) as our evaluation metrics. For a comprehensive evaluation of performance, we also report the true detection rate (TDR) for a given false detection rate (FDR).
Implementation Details. For data preprocessing, we only resize the facial images into a fixed size of 224×224. For training, we set the size of the mini-batch to 128, and the ratio of real and fake images to $1:1$. We use Adam [19] as our optimizer with an initial learning rate of 0.001 and a half decay every 5000 iters. The maximum iters number is 30000. And we set λ_1 to λ_5 in Equation 10 as 1, 0.01, 1, 0.01 and 0.01. All the code is based on the PyTorch framework and trained with NVIDIA GTX 1080Ti.

4.2 Evaluations

To evaluate our method comprehensively, in this section, we perform in-dataset, cross-method and cross-dataset evaluation to demonstrate the generalizability and robustness of our method.
In-Dataset Evaluation. In-Dataset evaluation reflects the ability of the network to fit the distribution of the dataset, as shown in Table 1. In general, with the help of our framework, the performance of both detectors and mainstream networks has been improved in different degrees, which fully proves the effectiveness and adaptability of our framework, Among them, our methods (ResNest-50 + Ours) achieve the state of the art on Deepfakes and NeuralTextures. Notably,

Table 1. In-Dataset evaluation (ACC (%)) on FF++ (LQ). We combine each forgery and real dataset in pairs to construct four sub-datasets, and evaluate the corresponding performance. AVG: the average performance of the four sub-datasets. Noting that results for some methods are from [36]. After embedding into our framework, all detectors achieve considerable performance gains and even outperform other methods.

Method	DF	FF	FS	NT	AVG
Steg.Features [16]	67.00	48.00	49.00	56.00	55.00
LD-CNN [12]	75.00	56.00	51.00	62.00	61.00
C-Conv [5]	87.00	82.00	74.00	74.00	79.25
CP-CNN [37]	80.00	62.00	59.00	59.00	65.00
MesoNet [3]	90.00	83.00	83.00	75.00	82.75
F^3-Net [36]	96.81	94.01	95.85	79.36	91.51
Gram-Net [27]	95.12	88.01	93.34	76.12	88.15
+ Ours	**95.67**	**89.06**	**94.01**	**76.96**	**88.93**
RFM [44]	95.42	91.24	93.60	79.83	90.02
+ Ours	**95.92**	**92.27**	**93.97**	**80.14**	**90.58**
ResNet-50 [18]	95.23	87.79	92.34	76.28	87.91
+ Ours	**95.43**	**88.94**	**93.99**	**77.19**	**88.89**
Xception [9]	95.36	91.94	93.55	78.32	89.79
+ Ours	**96.50**	**93.62**	**94.76**	**79.02**	**90.98**
ResNest-50 [49]	95.98	92.16	93.13	78.22	89.87
+ Ours	**98.95**	**94.32**	**94.56**	**80.46**	**92.10**

for Deepfakes, we outperform the F^3-Net [36] and baseline by 2.14% and 2.97% in terms of ACC score. Although our best performance is still slightly worse than F^3-Net on FaceSwap, it is understandable because our method does not pursue a magical modification of the network architecture.

Cross-Method Evaluation. Forgery techniques are constantly iterating, and we need to address not only existing forgery methods, but also the most cutting-edge ones. Table 2 shows our method is superior to the baseline in most cases, but the performance of both methods will drop greatly in cross-method evaluation, which is inevitable, because the extremely strong feature extraction capability of convolutional networks leads to the overfitting of detectors. Our method only mitigates the degree of overfitting to a certain extent. but does not significantly improve the generalization performance.

Cross-Dataset Evaluation. Due to the differences in raw data and experimental details, there can be huge gaps in the distribution between different datasets corresponding to even the same method. As shown in Table 3, regardless of the method, the performance drops significantly when testing on the Celeb-DF dataset, which implies that the difference in the distribution of different datasets for the same method does exist. With the assistance of our framework, the performance of each backbone on FF++ is slightly improved, but the improvement on Celeb-DF is significant. Specifically, our method (ResNest-50+Ours) has a 14.38% improvement on Celeb-DF, while the improvement on FF++-DF is only 2.97%. Furthermore, our method (ResNest-50+Ours) outperforms the state-of-the-art results (Chen *et al.* [8]) by 4.12% in terms of AUC score. Among the methods using Xception as the backbone, our method also surpasses others.

Table 2. Cross-Method evaluation (AUC (%)) on FF++ (C40). We adopt Xception [9], which is widely used in face forgery detection, as a baseline for comparison on FF++. Specifically, we use one of the sub-datasets for training, and the rest for testing.

Train Set	Method	Test Set (AUC(%))				
		DF	FF	FS	NT	AVG
DF	Xception	99.21	58.81	64.79	59.69	70.63
	+ Ours	**99.22**	**60.18**	**68.19**	**61.17**	**72.19**
FF	Xception	66.39	95.40	56.58	57.59	68.99
	+ Ours	**67.13**	**96.07**	**61.36**	**59.98**	**71.14**
FS	Xception	80.00	56.65	94.55	53.42	71.16
	+ Ours	**82.68**	**56.77**	**94.76**	**54.23**	**72.11**
NT	Xception	**69.94**	**67.88**	57.59	86.72	**70.53**
	+ Ours	68.39	65.40	**58.34**	**87.89**	70.01

Table 3. Cross-Dataset evaluation on Celeb-DF (AUC (%)) by training on FF++-DF (ACC (%)). Our method outperforms all the methods with the same backbone (Xception) and achieves the best performance with the backbone of ResNest-50.

BackBone	Method	FF++-DF (Train)	Celeb-DF (Test)
Xception	F^3-Net [36]	97.97	65.17
Efficient-B4	Zhao *et al.* [53]	-	67.44
HRNet	Face X-ray [31]	-	74.76
Xception	SPSL [25]	96.91	76.88
-	Chen *et al.* [8]	98.84	78.26
ResNet-18	Gram-Net [27]	95.12	67.14
	+ Ours	**95.67**	**74.94**
Xception	RFM [44]	95.42	67.21
	+ Ours	**95.92**	**74.44**
ResNet-50	ResNet-50 [18]	95.23	66.84
	+ Ours	**95.43**	**74.71**
Xception	Xception [9]	95.36	65.50
	+ Ours	**96.50**	**76.91**
ResNest-50	ResNest-50 [49]	95.98	68.00
	+ Ours	**98.95**	**82.38**

4.3 Ablation Study

We perform several ablations to better understand the contributions of each component in our method, the experimental results and visualizations are shown in Table 4 and Figure 3, respectively.

From the comparison of Variant A and Baseline, we can find that the performance of face forgery detection does not increase but decreases (0.81%) by simply introducing the disentanglement framework. Furthermore, we add *Content Consistency Constraint* (C^2C) and *Global Representation Contrastive Constraint* (GRCC) separately, with 4.97% and 7.46% improvement in terms of AUC, respectively, which proves the effectiveness of the enhanced independence of disentangled features. While the performance increases by 11.80% after combining these two, which indicates the two can play a mutually reinforcing role. Overall, C^2C and GRCC play a dominant role as the key core of our method.

Table 4. Ablation study on the FF++-DF and Celeb-DF. "Basic" represents the basic disentanglement framework.

Method	Basic	C^2C	GRCC	FF++-DF	Celeb-DF
Xception				95.36	65.50
Variant A	✓			94.55	65.11
Variant B	✓	✓		95.73	70.08
Variant C	✓		✓	96.33	72.57
Variant D	✓	✓	✓	**96.50**	**76.91**

Table 5. Results (Δ_{AUC} (%)) of image- and feature-level data augmentation study.

Method	Augmentation			Dataset (Δ_{AUC} (%))	
	Erasing [55]	H-Flip	Mixup [50]	FF++-DF	Celeb-DF
Image	✓			-2.13	**+0.98**
Feature	✓			**+0.34**	+0.94
Image		✓		-0.10	+0.23
Feature		✓		**+0.22**	**+3.62**
Image			✓	-0.85	+3.01
Feature			✓	**-0.07**	**+3.57**

4.4 Augmentation Study of Disentangled Features

Our framework first disentangles content features and artifact features from the images, and then uses the artifact features for subsequent detection. It is natural to guess that compared to image-level data augmentation, directly performing data augmentation on artifact features may achieve better performance. To validate it, we select common data augmentation methods such as Random Erasing [55], Horizontal Flip, and Mixup [50] to experiment, the details of the augmentation are shown in Figure 4. It is worth noting that data augmentation is not used in other experiments.

It can be seen from Table 5 that the performance improvement in the cross-dataset evaluation is greater than in-dataset evaluation. Our explanation is that in the in-dataset evaluation, the distributions of the train and test sets are close, and data augmentation disrupts the consistency of the distribution of the train and test set, resulting in little performance improvement or even reduction. For cross-dataset evaluation, data augmentation can enhance the diversity of the train set, and then pull the distribution between the train and test sets. In addition, the performance improvement of data augmentation at the feature-level is significantly better than that at the image-level, which implies the effectiveness of our disentanglement framework.

4.5 Investigation of Intrinsic Content Bias

To investigate the impact of intrinsic content bias within the dataset on the performance of face forgery detection, we cleverly construct two unbalanced datasets based on the FF++ dataset, the *Identity Unbalanced* dataset and the *Backgroud Unbalanced* dataset.

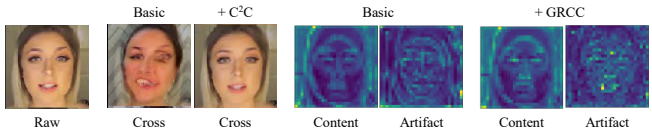


Fig. 3. Visualization of the ablation study, which illustrates the impact of C^2C on the reconstructed images and GRCC on the disentangled features, respectively."Raw" represents the raw image, and "Cross" represents the cross-reconstruction image.

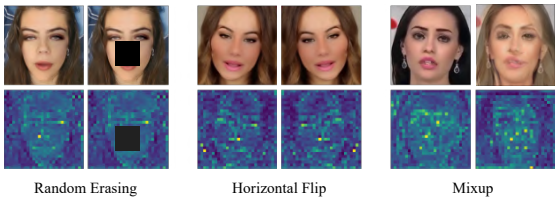


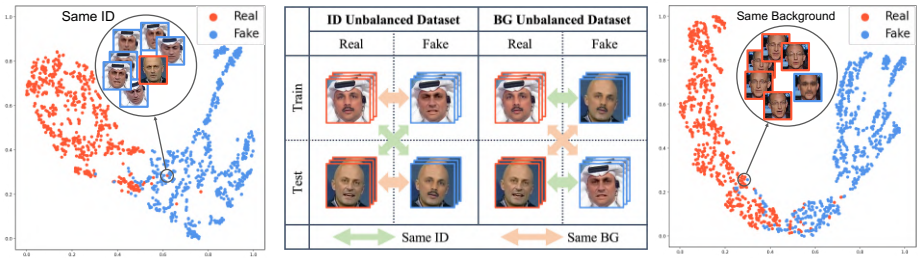
Fig. 4. Visualization of the image- (1st row) and feature-level (2nd row) augmentation.

We conduct comparative experiments on these datasets, and the experimental results are shown in Table 6. We can find that the performance on the ID and BG unbalanced datasets suffers a huge drop, which indicates that the existence of the intrinsic bias does interfere with the optimization of the detector. In contrast, our framework can maintain a high performance even on the unbalanced dataset by stripping the content features and thus eliminating the interference of content bias. Furthermore, compared with the ID unbalanced dataset, the performance degradation on the BG unbalanced dataset is more serious.

For a more intuitive understanding of the impact of content bias, we also visualize the t-SNE [29] feature spaces of the Xception network on the ID unbalanced dataset (Figure 5 (a)) and BG unbalanced dataset (Figure 5 (c)). We

Table 6. Comparison of our framework with baseline methods on the identity and background unbalanced dataset.

Method	ID Unbalanced Dataset				BG Unbalanced Dataset			
	ACC	AUC	EER	$TDR_{0.1}$	ACC	AUC	EER	$TDR_{0.1}$
Gram-Net [27]	89.85	96.49	10.14	89.70	79.84	87.83	20.19	66.10
+ Ours	**94.80**	**99.48**	**3.619**	**98.90**	**94.00**	**98.93**	**5.764**	**96.70**
RFM [44]	90.34	95.34	9.232	90.93	85.09	92.33	14.89	79.58
+ Ours	**95.49**	**99.11**	**4.102**	**97.90**	**95.02**	**98.34**	**4.839**	**96.93**
ResNet-50 [18]	89.61	96.46	10.35	89.30	80.88	91.29	17.17	72.30
+ Ours	**95.39**	**99.54**	**3.571**	**99.00**	**94.46**	**98.76**	**5.524**	**97.20**
Xception [9]	91.06	96.91	8.967	91.50	84.39	92.42	17.17	78.10
+ Ours	**95.85**	**99.32**	**3.434**	**99.10**	**95.14**	**98.71**	**4.762**	**97.00**
ResNest-50 [49]	89.89	97.54	8.507	92.70	81.28	93.68	14.34	79.60
+ Ours	**95.58**	**99.61**	**3.190**	**98.90**	**94.56**	**98.48**	**5.479**	**96.90**



(a) t-SNE of ID unbalanced dataset (b) Construction process of unbalanced datasets (c) t-SNE of BG unbalanced dataset

Fig. 5. (b) The construction process of unbalanced datasets. (a)(c) t-SNE feature visualization of the Xception network on the ID and BG unbalanced dataset.

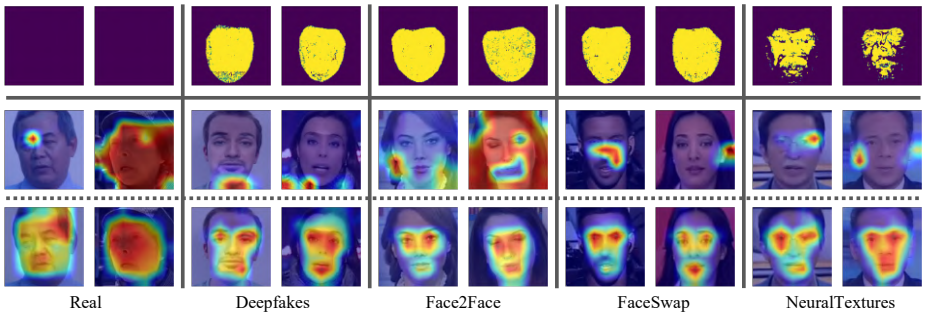
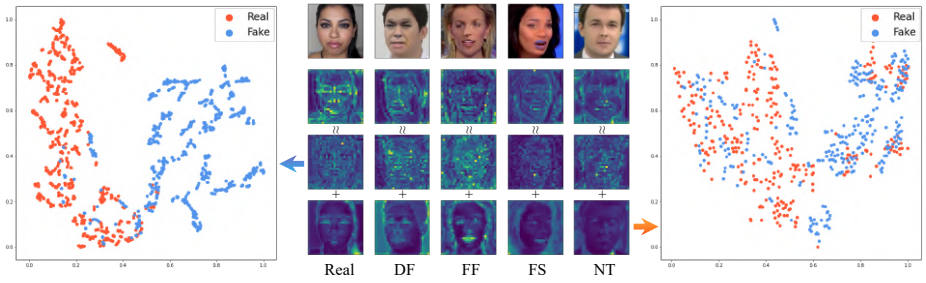


Fig. 6. Visualization of forgery mask (first row), Xception's (second row) and ours (third row) Grad-CAM on five sub-datasets of FF++. The activation region of our method is comprehensive and almost consistent with the forgery mask.

can observe that some samples with similar content information tend to cluster together, in other words, the distance between some samples with similar content information is much smaller than the distance between samples with similar forgery methods, which reveals that the content bias induce the detector to use content information for discrimination instead of artifact traces.

4.6 Visualization

To more intuitively demonstrate the effectiveness of our method, we visualize the Grad-CAM [39] of the baseline and our method, respectively, and the forgery mask, as shown in Figure 6. Grad-CAM shows that the baseline is prone to overfitting to small local regions or focusing on content noise outside the forgery region. In contrast, the activation region of our method is comprehensive and almost consistent with the forgery mask. Such visualization results also explain the motivation of this paper: without additional constraints, the detector has difficulty in mining suspicious artifact regions thorough weak supervision of labels only, and easily falls into content bias, which leads to overfitting or even



(a) t-SNE of artifact features (b) Visualization of images and features (c) t-SNE of content features

Fig. 7. (b) Visualization of the image (first row), traditional detector's (Xception) features (second row), ours disentangled artifact (third row) and content features (fourth row). (a)(c) t-SNE visualization of artifact features and content features.

misleading the direction of optimization. Instead, our goal is to remove the interference of content bias by an pre-disentanglement framework, and guide the detector to mine suspicious artifact trace.

As shown in the Figure 7 (b), traditional methods seek to allocate more attention to the face region, which improve the fitting ability but also exacerbated the overfitting of content bias within the dataset. Instead, we separate content features to eliminate misleading content information, guide the detector to pay attention to suspicious artifact traces, and strengthen the generalization capability fundamentally. Furthermore, Figure 7 (a)(c) demonstrate that the disentangled artifact features are discriminative for forgery detection, while the content features do not, which also validates the validity of our motives.

5 Conclusion

In this paper, we observe that detectors may no longer mine hard-to-capture artifact traces, and instead overfit certain content information, thus leading to the failure of generalization, which brings a novel perspective for face forgery detection. Motivated by this key observation, we design an easily embeddable disentanglement framework for content information removal, and further propose a *Content Consistency Constraint* (C^2C) and a *Global Representation Contrastive Constraint* (GRCC) to enhance the independence of disentangled features. Furthermore, we cleverly construct two unbalanced datasets to investigate the impact of the content bias. Extensive visualizations and experiments demonstrate our framework can not only ignore the interference of content bias but also guide the detector to mine suspicious artifact traces and achieve competitive performance in face forgery detection.

Acknowledgments. This work was supported by National Key R&D Program of China (2019YFB1406504).

References

1. Deepfake porn is ruining women's lives. https://www.technologyreview.com/2021/02/12/1018222/deepfake-revenge-porn-coming-ban, accessed: 2021-08-15
2. A voice deepfake was used to scam a ceo out of $243,000. https://www.forbes.com/sites/jessedamiani/2019/09/03/a-voice-deepfake-was-used-to-scam-a-ceo-out-of-243000, accessed: 2021-02-01
3. Afchar, D., Nozick, V., Yamagishi, J., Echizen, I.: Mesonet: a compact facial video forgery detection network. In: 2018 IEEE International Workshop on Information Forensics and Security (WIFS). pp. 1–7. IEEE (2018)
4. Asnani, V., Yin, X., Hassner, T., Liu, X.: Reverse engineering of generative models: Inferring model hyperparameters from generated images. arXiv preprint arXiv:2106.07873 (2021)
5. Bayar, B., Stamm, M.C.: A deep learning approach to universal image manipulation detection using a new convolutional layer. In: Proceedings of the 4th ACM workshop on information hiding and multimedia security. pp. 5–10 (2016)
6. Bengio, Y., Courville, A., Vincent, P.: Representation learning: A review and new perspectives. IEEE transactions on pattern analysis and machine intelligence **35**(8), 1798–1828 (2013)
7. Chai, L., Bau, D., Lim, S.N., Isola, P.: What makes fake images detectable? understanding properties that generalize. In: European Conference on Computer Vision. pp. 103–120. Springer (2020)
8. Chen, S., Yao, T., Chen, Y., Ding, S., Li, J., Ji, R.: Local relation learning for face forgery detection. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 35, pp. 1081–1088 (2021)
9. Chollet, F.: Xception: Deep learning with depthwise separable convolutions. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1251–1258 (2017)
10. Chugh, K., Gupta, P., Dhall, A., Subramanian, R.: Not made for each other-audio-visual dissonance-based deepfake detection and localization. In: Proceedings of the 28th ACM International Conference on Multimedia. pp. 439–447 (2020)
11. Ciftci, U.A., Demir, I., Yin, L.: Fakecatcher: Detection of synthetic portrait videos using biological signals. IEEE Transactions on Pattern Analysis and Machine Intelligence (2020)
12. Cozzolino, D., Poggi, G., Verdoliva, L.: Recasting residual-based local descriptors as convolutional neural networks: an application to image forgery detection. In: Proceedings of the 5th ACM Workshop on Information Hiding and Multimedia Security. pp. 159–164 (2017)
13. Dang, H., Liu, F., Stehouwer, J., Liu, X., Jain, A.K.: On the detection of digital face manipulation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern recognition. pp. 5781–5790 (2020)
14. deepfakes: Deepfakes. https://github.com/deepfakes, accessed: 2021-08-18
15. Frank, J., Eisenhofer, T., Schönherr, L., Fischer, A., Kolossa, D., Holz, T.: Leveraging frequency analysis for deep fake image recognition. In: International Conference on Machine Learning. pp. 3247–3258. PMLR (2020)
16. Fridrich, J., Kodovsky, J.: Rich models for steganalysis of digital images. IEEE Transactions on Information Forensics and Security **7**(3), 868–882 (2012)
17. Haliassos, A., Vougioukas, K., Petridis, S., Pantic, M.: Lips don't lie: A generalisable and robust approach to face forgery detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5039–5049 (2021)

18. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 770–778 (2016)
19. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)
20. 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: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 6458–6467 (2021)
21. Li, L., Bao, J., Zhang, T., Yang, H., Chen, D., Wen, F., Guo, B.: 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 (2020)
22. Li, Y., Chang, M.C., Lyu, S.: In ictu oculi: Exposing ai created fake videos by detecting eye blinking. In: 2018 IEEE International Workshop on Information Forensics and Security (WIFS). pp. 1–7. IEEE (2018)
23. Li, Y., Yang, X., Sun, P., Qi, H., Lyu, S.: 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 (2020)
24. Liang, J., Deng, W.: Identifying rhythmic patterns for face forgery detection and categorization. In: 2021 IEEE International Joint Conference on Biometrics (IJCB). pp. 1–8. IEEE (2021)
25. 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: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 772–781 (2021)
26. Liu, Y., Stehouwer, J., Liu, X.: On disentangling spoof trace for generic face anti-spoofing. In: European Conference on Computer Vision. pp. 406–422. Springer (2020)
27. Liu, Z., Qi, X., Torr, P.H.: Global texture enhancement for fake face detection in the wild. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 8060–8069 (2020)
28. Luo, Y., Zhang, Y., Yan, J., Liu, W.: Generalizing face forgery detection with high-frequency features. In: ProceeRings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 16317–16326 (2021)
29. Van der Maaten, L., Hinton, G.: Visualizing data using t-sne. Journal of machine learning research **9**(11) (2008)
30. MarekKowalski: Fakeswap. `https://github.com/MarekKowalski/FaceSwap`, accessed: 2021-08-18
31. Masi, I., Killekar, A., Mascarenhas, R.M., Gurudatt, S.P., AbdAlmageed, W.: Two-branch recurrent network for isolating deepfakes in videos. In: European Conference on Computer Vision. pp. 667–684. Springer (2020)
32. Matern, F., Riess, C., Stamminger, M.: Exploiting visual artifacts to expose deepfakes and face manipulations. In: 2019 IEEE Winter Applications of Computer Vision Workshops (WACVW). pp. 83–92. IEEE (2019)
33. Niu, X., Yu, Z., Han, H., Li, X., Shan, S., Zhao, G.: Video-based remote physiological measurement via cross-verified feature disentangling. In: European Conference on Computer Vision. pp. 295–310. Springer (2020)
34. Oord, A.v.d., Li, Y., Vinyals, O.: Representation learning with contrastive predictive coding. arXiv preprint arXiv:1807.03748 (2018)

35. Qi, H., Guo, Q., Juefei-Xu, F., Xie, X., Ma, L., Feng, W., Liu, Y., Zhao, J.: Deeprhythm: Exposing deepfakes with attentional visual heartbeat rhythms. In: Proceedings of the 28th ACM International Conference on Multimedia. pp. 4318–4327 (2020)
36. Qian, Y., Yin, G., Sheng, L., Chen, Z., Shao, J.: Thinking in frequency: Face forgery detection by mining frequency-aware clues. In: European Conference on Computer Vision. pp. 86–103. Springer (2020)
37. Rahmouni, N., Nozick, V., Yamagishi, J., Echizen, I.: Distinguishing computer graphics from natural images using convolution neural networks. In: 2017 IEEE Workshop on Information Forensics and Security (WIFS). pp. 1–6. IEEE (2017)
38. Rossler, A., Cozzolino, D., Verdoliva, L., Riess, C., Thies, J., Nießner, M.: Faceforensics++: Learning to detect manipulated facial images. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 1–11 (2019)
39. Selvaraju, R.R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., Batra, D.: Grad-cam: Visual explanations from deep networks via gradient-based localization. In: Proceedings of the IEEE international conference on computer vision. pp. 618–626 (2017)
40. Sun, K., Liu, H., Ye, Q., Liu, J., Gao, Y., Shao, L., Ji, R.: Domain general face forgery detection by learning to weight. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 35, pp. 2638–2646 (2021)
41. Thies, J., Zollhöfer, M., Nießner, M.: Deferred neural rendering: Image synthesis using neural textures. ACM Transactions on Graphics (TOG) **38**(4), 1–12 (2019)
42. Thies, J., Zollhofer, M., Stamminger, M., Theobalt, C., Nießner, M.: Face2face: Real-time face capture and reenactment of rgb videos. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2387–2395 (2016)
43. Tran, L., Yin, X., Liu, X.: Disentangled representation learning gan for pose-invariant face recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1415–1424 (2017)
44. Wang, C., Deng, W.: Representative forgery mining for fake face detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 14923–14932 (2021)
45. Wang, R., Juefei-Xu, F., Ma, L., Xie, X., Huang, Y., Wang, J., Liu, Y.: Fakespotter: A simple yet robust baseline for spotting ai-synthesized fake faces. In: Bessiere, C. (ed.) Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI-20. pp. 3444–3451. International Joint Conferences on Artificial Intelligence Organization (7 2020). https://doi.org/10.24963/ijcai.2020/476, `https://doi.org/10.24963/ijcai.2020/476`, main track
46. Wang, S.Y., Wang, O., Zhang, R., Owens, A., Efros, A.A.: Cnn-generated images are surprisingly easy to spot... for now. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 8695–8704 (2020)
47. Yang, X., Li, Y., Lyu, S.: Exposing deep fakes using inconsistent head poses. In: ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). pp. 8261–8265. IEEE (2019)
48. Yu, N., Davis, L.S., Fritz, M.: Attributing fake images to gans: Learning and analyzing gan fingerprints. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 7556–7566 (2019)
49. Zhang, H., Wu, C., Zhang, Z., Zhu, Y., Lin, H., Zhang, Z., Sun, Y., He, T., Mueller, J., Manmatha, R., et al.: Resnest: Split-attention networks. arXiv preprint arXiv:2004.08955 (2020)
50. Zhang, H., Cisse, M., Dauphin, Y.N., Lopez-Paz, D.: mixup: Beyond empirical risk minimization. In: International Conference on Learning Representations (2018)

51. Zhang, K.Y., Yao, T., Zhang, J., Tai, Y., Ding, S., Li, J., Huang, F., Song, H., Ma, L.: Face anti-spoofing via disentangled representation learning. In: European Conference on Computer Vision. pp. 641–657. Springer (2020)
52. Zhang, Z., Tran, L., Yin, X., Atoum, Y., Liu, X., Wan, J., Wang, N.: Gait recognition via disentangled representation learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 4710–4719 (2019)
53. Zhao, H., Zhou, W., Chen, D., Wei, T., Zhang, W., Yu, N.: Multi-attentional deepfake detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2185–2194 (2021)
54. Zhao, T., Xu, X., Xu, M., Ding, H., Xiong, Y., Xia, W.: Learning to recognize patch-wise consistency for deepfake detection. arXiv preprint arXiv:2012.09311 (2020)
55. Zhong, Z., Zheng, L., Kang, G., Li, S., Yang, Y.: Random erasing data augmentation. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 34, pp. 13001–13008 (2020)