



Improving the generalization of face forgery detection via single domain augmentation

Wenlong Li^{1,2} · Chunhui Feng^{1,2}  · Lifang Wei^{1,2} · Dawei Wu^{1,2}

Received: 8 May 2023 / Revised: 26 November 2023 / Accepted: 7 December 2023 /
Published online: 8 January 2024

© The Author(s), under exclusive licence to Springer Science+Business Media, LLC, part of Springer Nature 2024

Abstract

In recent years, significant progress has been made in the field of face forgery and face forgery detection. However, the performance of the detection methods in the unknown environment is far beyond satisfactory due to the feature distribution deviation of different fake face generators. In this paper, we adopt the domain generalization theory to improve the generality of fake face detection. The utilized method augments the original image samples by introducing gradient noise yielded during back-propagation, simulating the forgery features in unknown domains. In the construction of the detection network, we propose a multi-scale synthetic artifact trace tracker (MSATT) to enhance the manipulation traces through multi-scale content suppression. Meanwhile, we observed that the synthesized images present a noticeable color abnormality after going through the proposed MSATT module. Therefore, we designed a color difference perception network (CDPNet) to capture this unique feature. Experimental results demonstrate that both the domain augmentation and the proposed CDPNet can effectively improve the performance of the detection network. The proposed method is competitive with the state-of-the-art face forgery detection methods on both intra- and inter-dataset evaluations.

Keywords Face forgery detection · Domain generalization · Multi-scale synthetic artifact · Color-difference feature

1 Introduction

The highly realistic fake human faces in digital images or videos synthesized by machine-learning systems (deepfakes) have caused broad concerns. Abuse of these generated faces poses considerable threats to social security, such as fake news and video scams. Therefore,

✉ Chunhui Feng
fengchunhui@fafu.edu.cn

¹ College of Computer and Information Sciences, Fujian Agriculture and Forestry University, Fuzhou 350002, China

² Center for Agroforestry Mega Data Science, School of Future Technology, Fujian Agriculture and Forestry University, Fuzhou 350002, China

it is urgent to develop effective detection techniques to ensure the credibility of multimedia information.

Significant achievements have been made in the field of face forgery detection in recent years. Some of these methods [1–5] were designed based on Convolution Neural Networks (CNN). Afchar et al. [1] used CNN to extract mesoscopic features for face forgery detection. Masi et al. [2] proposed a two-branch network structure. One branch propagates the original information, while the other branch suppresses the face content yet amplifies multi-band frequencies using a Laplacian of Gaussian (LoG) as a bottleneck layer. Nguyen et al. [3] adopted a novel capsule network for detecting forged images and videos. Liu et al. [4] combined spatial image and phase spectrum to capture the up-sampling artifacts of face forgery to improve the transferability for face forgery detection. Zhou et al. [6] proposed a two-stream network structure. One stream to detect tampering artifacts for classification. The other stream trains a patch-based triplet network to leverage features capturing local noise residuals and camera characteristics as a second stream. Zhao et al. [6] defined face forgery detection as a fine-grained classification task for the first time and proposed a multi-attentional deepfake detection network. Guo et al. [7] suppressed the content features to make the model pay more attention to the artifact regions. Kohli et al. [8] proposed a light weight 3DCNN for face forgery detection to increase efficiency. What's more, Kohli et al. [9] is used for facial forgery detection by searching for operational traces in the frequency. These methods can reach high accuracy on hold-out test sets, but their performance usually drops significantly on unseen domains.

A pressing need for deepfake forensics is improving the detectors' generality. In this paper, we incorporate the domain generalization theory to address this issue. To the best of our knowledge, the application of domain generalization theory in face forgery detection has yet to be fully discussed. Few works use domain generalization theory in deepfake detection. For example, the authors of [10] enrich the diversity of fake samples by giving a pool of the forgery configuration. In contrast, our method improves the generality of the detection model by introducing adaptive gradient noise to the training data. The additional gradient noise is generated during backpropagation, which effectively uses the information learned by the network itself. It is worth noting that the training phase only uses forged samples from a single domain (single domain refers to the feature space of samples generated using a single forgery method) and training in a single domain is challenging. Since the enhanced domain is generated under the premise that the worst-case scenario is satisfied, it will be constrained by semantic consistency [11], resulting in poor domain transmission, we adopt semantic constraints relaxing module used to relax semantics constraints during domain augmentation.

The statistically based deepfake artifacts usually exist in the form of subtle traces, which may be obscured by the media content. Accordingly, we design an MSATT module to suppress content features and enhance subtle manipulation traces in a multi-scale fashion. Furthermore, we observed that there exists color difference in images before and after the MSATT pre-processing, and the color difference is significantly more evident in the synthesized samples than in the real samples. In light of this phenomenon, we propose a CDPNet to capture this color-based feature to enhance the discriminability of the detection method.

The key contributions of this paper are threefold as below:

1. We introduce the domain augmentation theory into fake face detections. A domain augmentation module automatically simulates the unknown feature domains through perturbations on the source domain, therefore enhancing the generality of the detection model.

2. We design an MSATT module to extract robust face forgery traces. This module enhances the subtle forgery traces by employing multi-scale feature fusion as well as content feature suppression.
3. We propose a novel color-difference feature for fake face detection. We observe that the MSATT also reveals a distinct color-behavior difference between the original and the synthesized images. We construct a CDPNet to capture this unique feature.

2 Proposed method

In this section, we introduce the proposed method in detail. We first give the overall of the proposed method in Section 2.1. Section 2.2 introduce the task module, which includes the MSATT and the CDPNet. Then, the domain augmentation strategy will be discussed in Section 2.3.

2.1 Overview

The overall architecture of the proposed method is shown in Fig. 1.

The framework mainly includes a task module and a domain augmentation module. In the task module, the RGB images to be tested are sent to the MSATT to capture the enhanced forgery traces. In addition, the high-frequency images are extracted from the RGB sequence using the SRM filter [12] to explore the noise information. We proposed a CDPNet to deal with the MSATT exposed color abnormality feature. Following are the backbone layers and a binary classifier. The domain augmentation module (DA) uses the method in [13] to enhance the source domain by simulating the distributions in the unknown environment under the worst-case constraint [11]:

$$\min_{\theta} \sup_{T: D(S,T) \leq \delta} \mathbf{E}[L_{task}(\theta; T)] \quad (1)$$

Where D is the similarity measurement between the original domain and the generalized target domain, and δ represents the largest domain boundary between Source Domain(S) and Target Domain(T). θ is a parameter optimized according to the objective function L of a specific task. In addition, we maximize domain transmission capacity with training with the assistance of the SCR module. The SCR module is a Wasserstein Auto-Encoders (WAEs) [14], which is used to relax the semantics constraints to generate more challenging samples. It is worth noting that the input images of the task module come from S and S^+ . S^+ means the enhanced domain generated by the domain augmentation module.

2.2 Task module

The task module extracts discriminative forgery features as well as trains a classifier. In each iteration, the samples from the original and enhancement domains are fed into the SRM filter and the MSATT module. The following backbone layers and the CDPNet extract discriminative features to train a classifier.

The pre-processing stage The task module starts with a two-branched pre-processing. The first branch adopts the SRM filter to extract the noise features. The other branch is the MSATT. In this paper, we mainly utilize the low-level statistical-based deepfake features to expose fake face forgery. According to [7], CNNs are more inclined to extract content representations

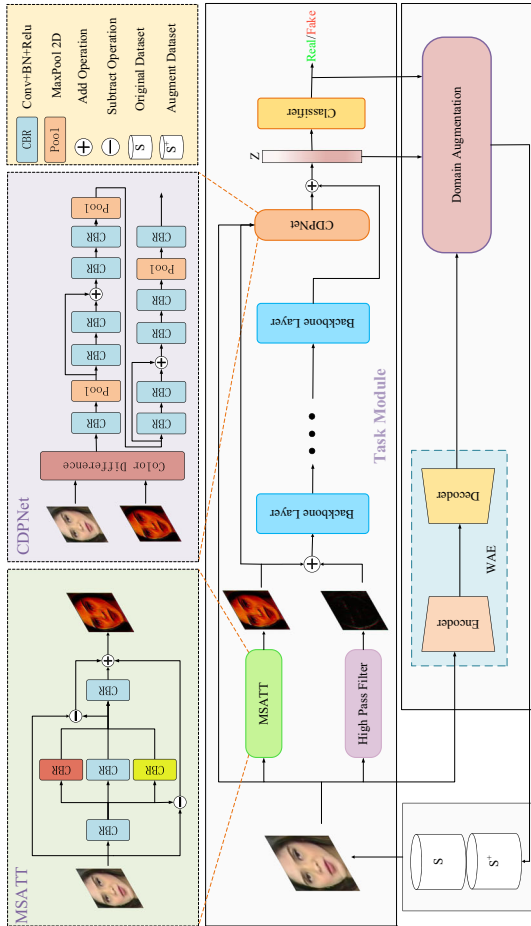


Fig. 1 The architecture of the proposed method

due to their relatively fixed structures. Moreover, the statistically based deepfake traces are often fragile and could be easily diminished. Therefore, we propose the MSATT to enhance the delicate forgery traces while suppressing the content features.

The MSATT is divided into three stages. In the first stage, a convolution operation $Conv(*)$ is used to calculate a feature map F_1 :

$$F_1 = Conv(\theta_1, x) \quad (2)$$

Where θ_1 denotes the parameters of the convolution layers in the first stage. $x \in \mathbb{R}^{W \times H \times C}$ denotes the input tensor. W denotes the height, H denotes the width, and C denotes the number of tensor channels.

The manipulation traces extracted using only one layer are fragile and could easily disappear in subsequent convolutions. Therefore in the second stage, a multi-scale convolution module is used to calculate multi-level feature maps F_2 from different perception fields.

$$F_2 = Conv_{<1 \times 1>}(\theta_2, F_1) + Conv_{<3 \times 3>}(\theta_2, F_1) + Conv_{<5 \times 5>}(\theta_2, F_1) \quad (3)$$

Where θ_2 denotes the parameters of multi-scale convolution layers in the second stage. $< * >$ indicates the different size of the convolution layer' filter.

In the third stage of the MSATT module, we subtract the input tensor x from F_1 and F_2 , respectively, to suppress the content representations while exposing the subtle manipulation traces. For further exposing as many discriminative cues for face forgery detection, we use a convolution layer in the third stage to perform convolution on the output F_2 of the second stage to fully utilize the remaining information. Meanwhile, we add the output of the high-frequency branch Q to get the final output tensor x' of the pre-processing stage:

$$x' = (F_1 - x) + (F_2 - x) + Q + Conv(\theta_3, F_2) \quad (4)$$

where θ_3 denotes the convolutional parameters of the third stage that preserves the additional discriminative information.

Then we use the backbone layers to obtain discriminative features and train a classifier. We use the cross-entropy loss function for the classification task:

$$L_{cls}(y, \hat{y}) = \frac{1}{N} \sum_i -[y_i \cdot \log(\hat{y}_i) + (1 - y_i) \cdot \log(1 - \hat{y}_i)] \quad (5)$$

where \hat{y}_i is the softmax output of the task model, y denotes the label, and N is the number of inputs.

Color difference perception network (CDPNet) After extensive observations, we found that the output image of MSATT often has an overall hue difference compared to the input image. The hue difference is more significant and stable in synthesized images than in pristine images. We speculate that this phenomenon stems from the inherent color distribution abnormalities present in deepfake images and videos. However, these abnormalities are often subtle and imperceptible to the human eye. After the MSATT module, the deepfake anomalies are amplified, making such hue differences apparent. We illustrate this phenomenon in Fig. 2.

Motivated by this observation, we designed a CDPNet to capture the color-based feature. We first use (6) to measure the color difference in the image before and after the MSATT pre-processing.

$$Diff_{(1,2)} = \sqrt{(I_{1,R} - I_{2,R})^2 + (I_{1,G} - I_{2,G})^2 + (I_{1,B} - I_{2,B})^2} \quad (6)$$

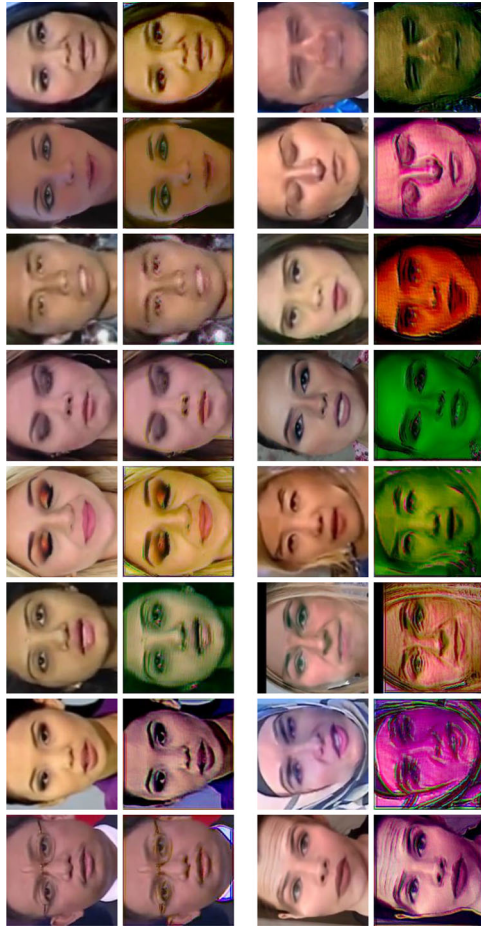


Fig. 2 The output images of MSATT. The first and third rows are the input pristine and forged images, respectively. The second and fourth rows are the images that expose the artifact trace after MSATT pre-processing

Table 1 The description of the convolution layers in the CDPNet

Layers	Layer Description	Output Size
layer1	<i>Conv</i> (3, 64, 7, 2)	128 × 128 × 64
layer2	<i>Max Pool2d</i> (64, 64, 3, 2)	64 × 64 × 64
layer3	<i>Conv</i> (64, 64, 1, 1)	64 × 64 × 64
layer4	<i>Conv</i> (64, 64, 3, 1)	64 × 64 × 64
layer5	<i>Conv</i> (64, 128, 1, 1)	64 × 64 × 128
layer6	<i>Conv</i> (128, 128, 3, 2)	32 × 32 × 128
layer7	<i>Max Pool2d</i> (128, 128, 3, 2)	16 × 16 × 128
layer8	<i>Conv</i> (128, 128, 1, 1)	16 × 16 × 128
layer9	<i>Conv</i> (128, 128, 3, 1)	16 × 16 × 128
layer10	<i>Conv</i> (128, 256, 1, 1)	16 × 16 × 256
layer11	<i>Conv</i> (256, 256, 3, 1)	16 × 16 × 256
layer12	<i>Max Pool2d</i> (256, 256, 3, 2)	8 × 8 × 256
layer13	<i>Conv</i> (256, 2048, 1, 1)	8 × 8 × 2048

where I_1 and I_2 represent the pixel value matrix of the input and output images of the MSATT module, respectively. R , G and B represent three color channels. After obtaining the color-difference measurement $Diff$, we fed it into the self-designed CDPNet. We only designed a few layers of convolutional structure and combined it with the short connections of ResNet to obtain more robust features. The structure of the CDPNet is listed in Table 1.

2.3 Domain augmentation

The purpose of domain augmentation is to make the detection model learn features from unknown distributions. The concept is shown in Fig. 3.

The domain augmentation simulates cross-domain distributions by adding gradient noise to the source domain. Due to the unpredictable nature of the noise signal, the augmentation expands in near-random directions based on the source domain. According to our observations and understanding of Deepfake datasets, we have found that the feature spaces of different

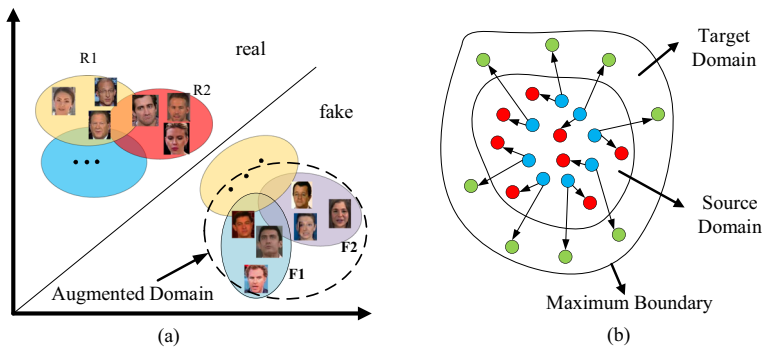


Fig. 3 The output images of MSATT. The first and third rows are the input pristine images and the forgery images respectively. The second and fourth rows are the images that expose the artifact trace after MSATT pre-processing

Deepfake datasets tend to cluster together. Therefore, despite the random nature of the domain augmentation, it still has the ability to extend to adjacent unknown domains.

We hope that the distribution of the enhancement domain should deviate from the source domain as much as possible to simulate a broader distribution of the unknown environment. This paper implements domain augmentation by satisfying the worst-case scenario under the constraint of ensuring semantic consistency, but the constraint of semantic consistency will limit the transmission capacity (the expansion capability from the source domain to the target domain), shown in Fig. 3(b), red spots generated from blue spots will be limited to a certain range. So we alleviate the constraints of semantic consistency while keeping it across the line between real and fake.

To achieve this goal, we exploit a specific domain augmentation strategy that simulates the unknown domain by introducing gradient noise into the source domain. The samples that need to be enhanced are considered as part of the trainable parameters of the task model and then use objective function L_{total} to calculate the gradients of the input layer to update the input samples, similar to the network's backpropagation process. L_{total} is the total loss required for the sample enhancement stage, as shown in (7).

$$L_{total} = L_{cls}(\theta; x) - \alpha L_{const}(\theta; z) + \beta L_{relax}(\varphi; x) \quad (7)$$

L_{total} consists of three losses: L_{cls} , L_{const} , and L_{relax} . α and β are hyperparameters to balance L_{const} and L_{relax} . We will detail these losses in the following.

L_{cls} is the classification loss defined in (5), it is the optimization function in the worst-case scenario.

L_{const} is used to maximize the difference between the source domain and the target domain while satisfying semantic consistency. Ensure that high-level semantic features are related to class labels. The definition is as follows:

$$L_{const} = \frac{1}{N} \sum_i \|z_i - z_i^+\|_2^2 + \infty \cdot 1\{y \neq y^+\} \quad (8)$$

where z^+ represents the discriminative features extracted from enhanced samples. $1\{\bullet\}$ indicate 0-1 indicator function and it will be ∞ if the class label of x^+ is different from x . L_{const} can

Table 2 The description of encoder and decoder

Layers	Layer Description	Output Size
Encoder		
layer1	Conv(3, 32, 3, 2)	128 × 128 × 32
layer2	2 × Conv(32, 64, 3, 2)	32 × 32 × 64
layer3	2 × Conv(64, 128, 3, 2)	8 × 8 × 128
layer4	2 × Conv(128, 256, 3, 2)	4 × 4 × 256
layer5	Conv(256, 512, 3, 2)	2 × 2 × 512
Decoder		
layer1	2 × ConvTranspose(512, 256, 3, 2)	8 × 8 × 256
layer2	2 × ConvTranspose(256, 128, 3, 2)	32 × 32 × 128
layer3	2 × ConvTranspose(128, 64, 3, 2)	128 × 128 × 64
layer4	ConvTranspose(64, 32, 3, 2)	256 × 256 × 32
layer5	ConvTranspose(32, 3, 3, 1)	256 × 256 × 3

achieve a certain degree of out-of-domain generalization in the embedded space, but its out-of-domain transmission ability is limited due to semantic consistency constraints. In order to enhance the transmission capability outside the domain and increase the diversity of samples, we adopted L_{relax} to alleviate the constraint of semantic consistency which is defined as follows.

L_{relax} is for mitigating constraints on semantic consistency by limiting the encoding capabilities of WAEs which are defined in (9).

$$L_{relax} = \frac{1}{N} \sum_i \|X_i^+ - D(E(X_i^+))\|_2^2 \quad (9)$$

Where E and D represent the Encoder and Decoder, respectively. In this paper, we use WAEs [14] to implement L_{relax} . By limiting the encoding ability of WAE, the reconstruction error increases, and more disturbances are generated to enhance the domain transmission capability. The structure of the encoder and decoder is described in Table 2. The encoder and decoder need to be trained in advance to better capture the distribution of the source domain. Then limit the encoding ability of WAEs to maximize the difference between the enhancement domain and the source domain.

Algorithm 1 The proposed method for training strategy.

Require: source domain S , pretrained WAEs V on S ; the number of iterations; the number of training batches

N ; the number of evaluate batches M ; the conditional of domain augmentation T

Ensure: task module parameter θ

```

1: for  $t = 1 \rightarrow iterations$  do
2:   if  $t \bmod T == 0$  then
3:     generate  $S_t^+$  from  $S_{t-1}$  using (10)
4:     retrain  $V$  with  $S_t^+$ 
5:     for  $i = 1 \rightarrow N$  do
6:       training task module with  $S_t = S_t^+ \cup S_{t-1}$ 
7:       update task module parameter  $\theta$ 
8:     end for
9:     for  $i = 1 \rightarrow M$  do
10:      evaluate task module with  $S_t$ 
11:    end for
12:   else
13:     for  $i = 1 \rightarrow N$  do
14:       training task module with  $S_{t-1}$ 
15:       update task module parameter  $\theta$ 
16:     end for
17:     for  $i = 1 \rightarrow M$  do
18:      evaluate task module with  $S_{t-1}$ 
19:    end for
20:   end if
21: end for

```

After obtaining the objective function L_{total} that needs to be optimized, we can use the iterative method to perturb the original sample along the direction of gradient change to generate more samples x^+ :

$$x_{t+1}^+ \leftarrow x_t^+ + \xi \nabla_{x_t^+} L_{total}(\theta, \varphi; x_t^+, z_t^+) \quad (10)$$

where ζ represents the scale factor. θ and φ represent the convolutional layers' parameters of the task module and SCR module respectively. Our main idea is to add disturbance to the sample to obtain new ones. The feature vector z is obtained from the samples in the source domain through the task model. In order to make the difference between z and the feature vector z^+ of the samples corresponding to the augment domain larger, we augment the samples along the gradient change direction in the back propagation process through adaptive learning. In simple terms, we update the input samples in a way similar to the network parameter update. It is also worth noting that during the non-data augmentation training phase, we only use L_{cls} for optimization of the parameters θ of the task module:

$$\hat{\theta} \leftarrow \theta - \eta \nabla_{\theta} L_{cls}(\theta; S \cup S^+) \quad (11)$$

Where η is the learning rate. Training task module on the original and enhanced domains to achieve better generalization performance.

Our work mainly includes the following two points. First, in order to better capture manipulation traces, MSATT is used to suppress content features to expose manipulation traces. Second, simulate the distribution outside the domain through the domain enhancement strategy. Using L_{const} and L_{relax} to maximize the expansion outside the domain by relaxing the constraints of semantic consistency. Our method implementation is summarized in Algorithm 1.

3 Experiment

In this section, we conduct several experiments to verify the effectiveness of the proposed method. Section 3.1 provides the details of the experimental setup. Section 3.2 reports the ablation experiment. Section 3.3 provides visualization of the generalization of our method. Section 3.4 reports the experimental results with recent works. Section 3.5 verifies the model generalization performance.

3.1 Experimental setup

Datasets We performed experiments on several of the most popular deepfake datasets: **FaceForensics++ (FF++)** [15], **Celeb-DF** [16], **Deepfake Detection Challenge (DFDC)** [17] and **FaceShifter** [18]. The FF++ contains four forgery patterns: **DeepFake (DF)** [19], **Face2Face (F2F)** [20], **FaceSwap (FS)** [21], and **NeuralTexture (NT)** [22]. A total of 4,000 forged videos were generated based on 1,000 pristine videos. In addition, according to different compression rates, FF++ also provides three different levels of compressed video: pristine quality (raw), high quality (HQ), and low quality (LQ). In this paper, We regard **DeepFake**, **Face2Face**, **FaceSwap**, and **NeuralTexture** as four datasets. Note that we divide the dataset according to the official ratio of 720:140:140. **Celeb-DF** [16] is another widely used deep forged dataset. It improves the visual quality of the video samples and is more challenging for face forgery detection tasks. **DeepFake Detection Challenge (DFDC)** [17] is another more challenging dataset which contains 1,000 pristine videos and over 4,000 fake videos manipulated by multiple DeepFake, GAN-based and non-learned methods. **FaceShifter** [18] is a more challenging face forgery detection that is not only considerably more perceptually appealing, but also better identity preserving in comparison.

Metrics The metrics in our experiments are Accuracy (ACC) and Area under the curve (AUC), which are most commonly used for evaluating face forgery detection methods.

Implementation details The backbone of the proposed architecture is the Xception [23] which is pre-trained on imagenet. We use MTCNN [24] to extract the face areas, and align and resize them to 256×256 pixels. The hyper-parameter used in (7) are $\alpha = 0.0001$ and $\beta = 1e9$. We set the batch size to 32, and use the Adam optimizer. The learning rate of the task module and WAE is set to 0.00002 and 0.0005, respectively. Our experiments run on an NVIDIA GTX GeForce 1080Ti GPU.

3.2 Ablation study

In this section, we carry out several ablation experiments to verify the effectiveness of the proposed MSATT, CDPNet, and Data Augmentation (DA) strategy. All ablation experiments were trained on DF and tested on each of the four sub-datasets in FF++. The results are shown in Table 3. The results show that the main modules we propose can improve the detection performance on both intra- and inter-datasets. The first row is the detection results of the backbone. In the second row, we tested the efficacy of the MSATT. Compared to the backbone, the performance slightly drops on the same domain but improves on the cross-domain.

In the third row, we evaluated the CDPNet. Since there is a dependency between the CDPNet and the MSATT, we tested the two modules jointly. Compared to the backbone, it has increased by 0.23%, 1.14%, 1.40%, and 0.57% on DF, F2F, FS, and NT, respectively. We analyzed the DA strategy in the fourth row. The results on cross-domain datasets have significantly improved in comparison to the baseline, which is evident that the DA enhances the generality of the detection algorithm. In the last row, we evaluated the performance after integrating all proposed modules. It demonstrates the optimal results both within and across datasets. Compared to the backbone, it has increased by 0.59%, 3.25%, 2.13%, and 1.11% on DF, F2F, FS and NT, respectively. We can see that the improvement in the intra-domain performance is not substantial, whereas there are notable improvements in cross-domain performances. The reason is that the baseline performance of the intra-domain (tested on DF) has already reached 99.14%, leaving little room for significant improvement. In contrast, there is ample room for improvement in cross-domain performances compared to the baseline. Moreover, the significant improvement in cross-dataset performance demonstrates the effectiveness of domain generalization.

Table 3 Ablation study on proposed modules(AUC(%))

ID	MSATT	CDPNet	DA	DF	F2F	FS	NT
1	-	-	-	99.14	63.22	58.32	67.61
2	✓	-	-	99.07	63.43	58.63	68.18
3	✓	✓	-	99.37	64.36	59.72	68.32
4	-	-	✓	99.65	65.35	60.12	68.65
5	✓	✓	✓	99.73	66.47	60.45	68.72

3.3 Visualization of improved generalization

The purpose of the proposed method is to improve the cross-domain generalization of the face forgery detection model.

We perturb the original samples to let the model learn more diverse representations under unknown distributions. Figure 4 demonstrates the visualization results of the feature domain augmentation. It realizes dimension reduction by using t-sne [25]. By comparing Fig. 4(a) and (b), we find that the feature distributions deviate further from the centroid after adding the DA. It means that our method does expand both the real and fake data domains. By observing (c) and (d), we find that after domain generalization, the model is better able to distinguish the real and fake data in the cross-dataset evaluation, which proves the DA can expand the feature domains along the correct directions to a certain extent.

3.4 In-Domain evaluation

In this section, we evaluate the in-domain performance of the proposed method. We compare our method with five SOTA deepfake forensic methods. All the methods are trained and tested

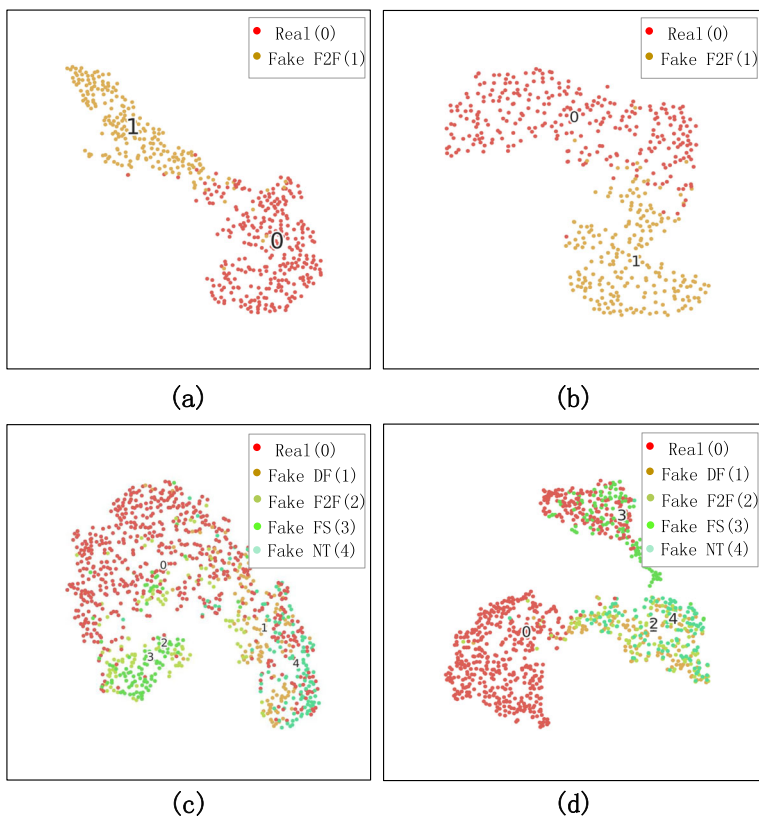


Fig. 4 The t-sne feature space visualization. Results of the first row demonstrate the feature distributions when the model is trained on F2F and tested on F2F. In the second row, the model is trained on F2F and tested on FF++. In the first and second columns, the detection model is trained without and with domain augmentation, respectively

Table 4 Comparative analysis of detection performance with other methods in seen domain

methods	DeepFake		Face2Face		FaceSwap		NeuralTexture		Average AUC
	ACC	AUC	ACC	AUC	ACC	AUC	ACC	AUC	
Afchar [1]	87.27	91.03	86.32	90.76	82.43	88.62	79.96	83.86	88.57
Chollet [23]	95.13	98.85	87.34	93.62	92.42	97.08	77.54	84.51	93.12
Qian [26]	-	98.62	-	98.51	-	98.34	-	93.22	97.17
Liu [4]	93.48	98.50	86.02	94.62	92.42	98.10	76.78	80.49	92.93
Yang [27]	-	98.96	-	98.90	-	97.37	-	95.65	97.72
Ours	96.43	99.68	97.60	99.79	97.65	99.74	90.31	99.64	99.71

on each of the four sub-datasets in FF++. Table 4 shows the detection results. We can see that the proposed method achieves competitive performances within the same domain. Our method’s AUC metric is 0.72% higher than that of Yang et al. [27] in the DeepFake subset. Our method’s AUC metric is 0.89% higher than that of Yang et al. [27] in the Face2Face. Testing on FaceSwap, our method’s AUC is 1.40% higher than that of Qian et al. [26]. Similarly, Testing on NeuralTexture, our method’s AUC is 3.99% higher than that of Yang et al. [27].

We show the comprehensive performance of all methods by averaging the AUC on four sub-datasets. We can see that our method is superior to other methods. Meanwhile, Fig. 5 visualize the ROC curve to show it more intuitively.

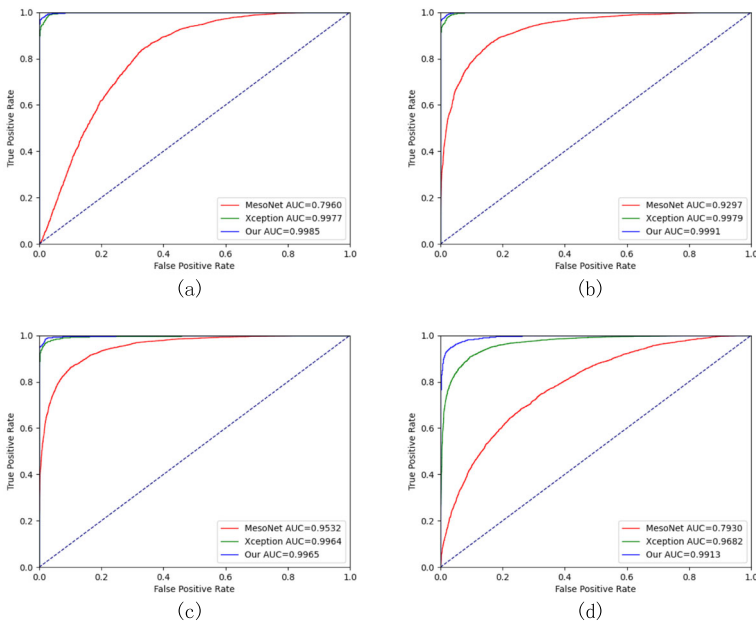


Fig. 5 (a) indicate training on DeepFake. (b) indicate training on Face2Face. (c) indicate training on FaceSwap. (d) indicate training on NeuralTexture

3.5 Out-of-domain evaluation

Cross-Manipulation Evaluation In this part, we evaluate the generalization of the proposed method to unseen manipulations. The datasets and results are shown in Table 5. For a fair comparison, the methods compared were all re-implemented so that the training sets could be kept the same. We can see that our method shows overall superior performance in comparison to the rest. Especially, on the training set F2F, our method is superior to all other methods on all the test sets. However, the proposed method does not perform adequately on the training set NT. Our analysis suggests that our method has a relatively ideal effect on the expression transfer method based on computer graphics, while the effect on deep learning expression transfer is not very satisfactory. We calculated the average AUC in the last column. Our results are respectively 1.74%, 3.53%, 4.36%, and 0.96% higher than the second-highest. We also provide histograms to intuitively demonstrate the generalization of the proposed method against the comparison methods, as in Fig. 6.

Table 5 Cross-Manipulation Evaluation(AUC(%)) of face forgery detection

train dataset	methods	test dataset				Average
		DF	F2F	FS	NT	
DF	Chollet [23]	99.14	63.22	58.32	67.61	72.07
	Tan [28]	99.57	66.32	46.32	69.35	70.37
	Afchar [1]	96.51	55.68	39.25	65.14	64.14
	Nguyen[3]	98.37	68.34	54.83	63.34	71.22
	Yu [29]	97.36	59.71	51.23	59.88	66.99
	ours	99.73	66.47	60.45	68.72	73.84
F2F	Chollet [23]	68.38	99.62	59.82	64.54	73.09
	Tan [28]	82.44	98.72	58.14	63.71	75.75
	Afchar [1]	65.07	92.64	60.33	64.24	70.57
	Nguyen[3]	76.58	99.31	71.34	64.96	78.04
	Yu [29]	71.66	96.47	61.35	60.83	72.57
	ours	82.74	99.79	75.64	68.12	81.57
FS	Chollet [23]	49.51	61.67	98.92	61.96	68.02
	Tan [28]	58.69	67.32	99.12	48.61	68.43
	Afchar [1]	61.12	62.23	93.41	62.21	69.74
	Nguyen[3]	47.36	63.87	99.21	64.38	68.70
	Yu [29]	56.37	65.21	98.36	52.12	68.01
	ours	63.34	76.68	99.74	56.67	74.10
NT	Chollet [23]	83.71	67.06	54.87	99.13	76.19
	Tan [28]	49.45	66.59	66.72	94.82	74.39
	Afchar [1]	67.47	61.49	54.55	82.37	66.46
	Nguyen[3]	76.31	66.12	51.26	97.34	72.76
	Yu [29]	59.32	59.68	51.79	92.17	75.74
	ours	87.06	62.82	59.09	99.64	77.15

The best and second-best results are marked in bold and red respectively

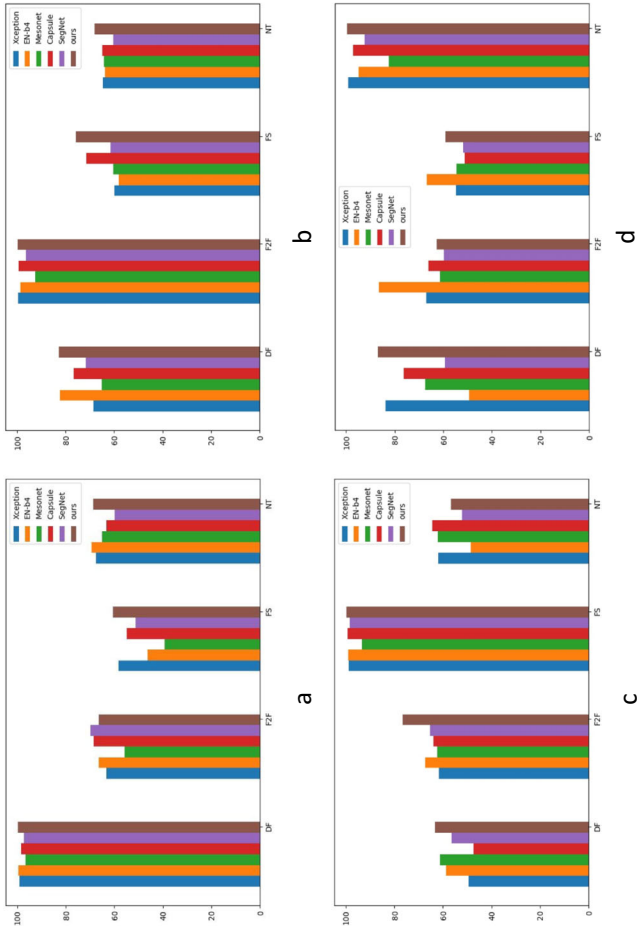


Fig. 6 (a) indicate training on DeepFake. (b) indicate training on FaceSwap. (c) indicate training on Face2Face. (d) indicate training on NeuralTexture

Cross-dataset evaluation In this part, we assess the cross-dataset generalization of the proposed method. The test datasets are Celeb-DF, DFDC, and FaceShifter. We train each model on DF, F2F, FS, and NT, respectively. We compared several SOTA deep face detection methods with good generalization performance. Li et al. [30] identify authenticity by detecting mixed boundaries. Qian. et al. [26] obtained identification clues through the frequency domain. The comparison results are shown in Table 6. Analyzing the experiment data can clearly find that our method can achieve the best or second-best results. Especially the results of training on DeepFake(DF), our method achieves the best test results on Celeb-DF, DFDC, and FaceShifter. This indicates that our method performs well in terms of generalization for forgery types of DeepFake. The training results on F2F and NT show that our method has satisfactory generalization performance on Celeb-DF and FaceShifter, and can also achieve the second best performance on DFDC. The effectiveness of our method on FS is not very satisfactory, possibly due to insufficient learning of this type of forgery in our method. In addition, we also calculated the average AUC of the four different manipulation test datasets. It can be intuitively seen that our method can achieve the best result on F2F, it is 2.71%

Table 6 Cross-Dataset Evaluation(AUC(%)) of face forgery detection

train dataset	methods	test dataset			
		Celeb-DF	DFDC	FaceShifter	Average
DF	Chollet [23]	63.48	61.72	58.96	63.54
	Afchar [1]	61.01	58.63	58.53	59.39
	Yu [29]	54.86	60.11	61.28	58.75
	Li [30]	55.41	66.80	60.91	61.04
	Qian [26]	66.40	65.80	68.21	66.81
	ours	68.72	69.89	69.48	69.36
F2F	Chollet [23]	65.14	74.51	70.81	70.15
	Afchar [1]	50.37	69.43	58.69	59.49
	Yu [29]	56.42	71.28	61.02	62.91
	Li [30]	68.41	79.60	63.31	70.44
	Qian [26]	65.40	76.10	67.91	69.81
	ours	69.74	77.65	73.88	73.75
FS	Chollet [23]	63.71	60.51	70.81	65.01
	Afchar [1]	60.93	55.89	64.84	60.55
	Yu [29]	61.38	59.31	66.38	62.35
	Li [30]	69.71	79.23	64.61	71.18
	Qian [26]	63.61	65.10	67.91	65.54
	ours	68.95	66.52	71.68	69.05
NT	Chollet [23]	64.92	83.81	64.61	71.11
	Afchar [1]	39.86	68.27	56.47	54.86
	Yu [29]	51.62	70.15	61.75	61.17
	Li [30]	70.03	85.92	61.31	72.42
	Qian [26]	68.90	93.21	67.21	76.44
	ours	72.81	87.12	68.25	76.06

The best and second-best results are marked in bold and red respectively

higher than that of the second-best result method. The other three results can achieve the second-best effect.

4 Conclusion

In this paper, we provide a new feasible method for improving the generality of face forgery detection. We adopt the domain generalization theory to simulate the real and fake face feature distributions in the unknown environment. The employed technique introduces gradient disturbances to the source domain in an automatic fashion. We demonstrate the improved generalization through visualizations and quantitative results. Since the domain extension is uncontrollable in direction, it is actually generalizing in a nearly-random manner. In the future, we will investigate methods that can control domain augmentation under meaningful ranges.

We find that when trying to enhance the subtle manipulation traces through specifically designed CNN structures, the outputs reveal chromatic anomalies, such as hue-shifts after the proposed MSATT module. This result supports that the colors of deepfake images are actually unnatural, which will become evident with magnification. In future research, we will continue to study these hidden features, such as color features of other attributions.

Acknowledgements This work was supported by the National Natural Science Foundation of China under Grant 61802064.

Data Availability The datasets generated during and analyzed during the current study are available from the corresponding author upon reasonable request.

Declarations

Conflict of Interests The authors declare that they have no conflict of interest.

References

1. Afchar D, Nozick V, Yamagishi J, Echizen I (2018) Mesonet: a compact facial video forgery detection network. In: 2018 IEEE international workshop on information forensics and security, pp 1–7. IEEE
2. Masi I, Killekar A, Mascarenhas, RM, Gurudatt, SP, AbdAlmageed W (2020) Two-branch recurrent network for isolating deepfakes in videos. In: European conference on computer vision, pp 667–684. Springer
3. Nguyen HH, Yamagishi J, Echizen I (2019) Capsule-forensics: using capsule networks to detect forged images and videos. In: ICASSP 2019-2019 IEEE international conference on acoustics, speech and signal processing (ICASSP), pp 2307–2311. IEEE
4. Liu H, Li X, Zhou W, Chen Y, He Y, Xue H, Zhang W, Yu N (2021) 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
5. Zhou P, Han X, Morariu VI, Davis LS (2017) Two-stream neural networks for tampered face detection. In: 2017 IEEE conference on computer vision and pattern recognition workshops (CVPRW), pp 1831–1839. IEEE
6. Zhao H, Zhou W, Chen D, Wei T, Zhang W, Yu N (2021) Multi-attentional deepfake detection. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 2185–2194
7. Guo Z, Yang G, Chen J, Sun X (2021) Fake face detection via adaptive manipulation traces extraction network. *Comput Vis Image Underst* 204:103170
8. Kohli A, Gupta A (2022) Light-weight 3dcnn for deepfakes, faceswap and face2face facial forgery detection. *Multimed Tool Appl* 81(22):31391–31403

9. Kohli A, Gupta A (2021) Detecting deepfake, faceswap and face2face facial forgeries using frequency cnn. *Multimed Tool Appl* 80:18461–18478
10. Chen L, Zhang Y, Song Y, Liu L, Wang J (2022) Self-supervised learning of adversarial example: towards good generalizations for deepfake detection. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp 18710–18719
11. Volpi R, Namkoong H, Sener O, Duchi JC, Murino V, Savarese S (2018) Generalizing to unseen domains via adversarial data augmentation. *Advan Neural Inform Process Syst* 31
12. Luo Y, Zhang Y, Yan J, Liu W (2021) Generalizing face forgery detection with high-frequency features. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp 16317–16326
13. Qiao F, Zhao L, Peng X (2020) Learning to learn single domain generalization. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp 12556–12565
14. Tolstikhin I, Bousquet O, Gelly S, Schoelkopf B (2017) Wasserstein auto-encoders. [arXiv:1711.01558](https://arxiv.org/abs/1711.01558)
15. Rossler A, Cozzolino D, Verdoliva L, Riess C, Thies J, Nießner M (2019) Faceforensics++: learning to detect manipulated facial images. In: *Proceedings of the IEEE/CVF international conference on computer vision*, pp 1–11
16. Li Y, Yang X, Sun P, Qi H, Lyu S (2020) 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
17. Dolhansky B, Bitton J, Pflaum B, Lu J, Howes R, Wang M, Ferrer CC (2020) The deepfake detection challenge (dfd) dataset. [arXiv:2006.07397](https://arxiv.org/abs/2006.07397)
18. Jiang L, Li R, Wu W, Qian C, Loy CC (2020) Deepforensics-1.0: a large-scale dataset for real-world face forgery detection. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp 2889–2898
19. Deepfakes github. website, <https://github.com/deepfakes/faceswap>. Accessed: 24 March 2022
20. Thies J, Zollhofer M, Stamminger M, Theobalt C, Nießner M (2016) 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
21. Faceswap (2018) <https://github.com/marekkowalski/faceswap/>. Accessed: 29 October 2018
22. Thies J, Zollhöfer M, Nießner M (2019) Deferred neural rendering: Image synthesis using neural textures. *Acm Transactions on Graphics (TOG)* 38(4):1–12
23. Chollet F (2017) Xception: deep learning with depthwise separable convolutions. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp 1251–1258
24. Xiang J, Zhu G (2017) Joint face detection and facial expression recognition with mtcn. In: *2017 4th International conference on information science and control engineering (ICISCE)*, pp 424–427. IEEE
25. Van der Maaten L, Hinton G (2008) Visualizing data using t-sne. *J Mach Learn Res* 9(11)
26. Qian Y, Yin G, Sheng L, Chen Z, Shao J (2020) Thinking in frequency: face forgery detection by mining frequency-aware clues. In: *European conference on computer vision*, pp 86–103. Springer
27. Yang J, Xiao S, Li A, Lu W, Gao X, Li Y (2021) Msta-net: forgery detection by generating manipulation trace based on multi-scale self-texture attention. *IEEE Trans Circ Syst Video Technol*
28. Tan M, Le Q (2019) Efficientnet: rethinking model scaling for convolutional neural networks. In: *International conference on machine learning*, pp 6105–6114. PMLR
29. Yu C-M, Chen K-C, Chang C-T, Ti Y-W (2022) Segnet: a network for detecting deepfake facial videos. *Multimedia Syst* 28(3):793–814
30. Li L, Bao J, Zhang T, Yang H, Chen D, Wen F, Guo B (2020) Face x-ray for more general face forgery detection. In: *Proceedings of the IEEE conference on CVPR*, pp 5001–5010

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.