Generalizing Face Forgery Detection with High-frequency Features

Yuchen Luo*†[1,2] Yong Zhang*[3] Junchi Yan‡[1,2] Wei Liu‡[4]
[1]Department of Computer Science and Engineering, Shanghai Jiao Tong University
[2]MoE Key Lab of Artificial Intelligence, AI Institute, Shanghai Jiao Tong University
[3]Tencent AI Lab [4]Tencent Data Platform
{592mcavoy,yanjunchi}@sjtu.edu.cn zhangyong201303@gmail.com wl2223@columbia.edu

Abstract

Current face forgery detection methods achieve high accuracy under the within-database scenario where training and testing forgeries are synthesized by the same algorithm. However, few of them gain satisfying performance under the cross-database scenario where training and testing forgeries are synthesized by different algorithms. In this paper, we find that current CNN-based detectors tend to overfit to method-specific color textures and thus fail to generalize. Observing that image noises remove color textures and expose discrepancies between authentic and tampered regions, we propose to utilize the high-frequency noises for face forgery detection. We carefully devise three functional modules to take full advantage of the high-frequency features. The first is the multi-scale high-frequency feature extraction module that extracts high-frequency noises at multiple scales and composes a novel modality. The second is the residual-guided spatial attention module that guides the low-level RGB feature extractor to concentrate more on forgery traces from a new perspective. The last is the cross-modality attention module that leverages the correlation between the two complementary modalities to promote feature learning for each other. Comprehensive evaluations on several benchmark databases corroborate the superior generalization performance of our proposed method.

1. Introduction

As face manipulation techniques [4, 24, 59] spring up along with the breakthrough of deep generators [27, 20], face forgery detection becomes an arousing research topic. Most existing methods focus on within-database detection [7, 29, 9], where forged images in the training set and testing set are manipulated by the same algorithm. However, the biggest challenge hampering face forgery detection is the generalization problem. Due to the diversified data distributions generated by different manipulation techniques, methods with high within-database detection accuracy always experience a severe performance drop in the cross-database scenario, thus limiting broader applications.

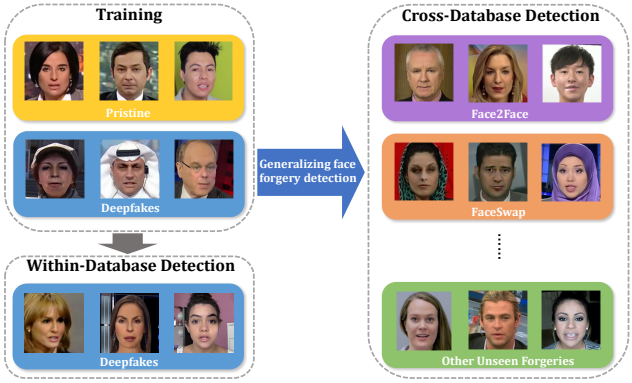

Figure 1: Training and testing forgeries of within-database detection are synthesized by the same algorithm while those of cross-database detection are synthesized by different algorithms. We focus on the latter which is more challenging.

Recently, several works are devoted to addressing the generalizing problem. [30, 28] assume that some artifacts are shared in forged faces and customize databases specialized on those artifacts. However, these assumptions do not always hold. Besides, transfer learning [13], domain adaptation [54], and multi-task learning [14, 35, 45] are utilized to improve model's performance in unseen domains. Nevertheless, the acquisition of target samples and annotations is expensive. Meanwhile, some attempt to obtain information from frequency domains, such as Fourier transformation [11], DCT [38], and steganalysis features [52, 57]. But they rarely consider the relation and interaction between the additional information and regular color textures.

In this paper, we aim at learning a more generalizable face forgery detector (See Fig. 1). To facilitate understanding why current CNN-based works fail on unseen forgeries, we analyze CNN-based classifiers' behaviors and find that

*Equal contribution
†Work done during an internship at Tencent AI Lab
‡Corresponding Author

the model is biased to method-specific color textures. Observing that high-frequency noises can suppress image textures and expose statistical discrepancies between tampered and authentic regions, we propose to utilize noises to tackle the overfitting problem.

We propose a generalizable model for face forgery detection. To take full advantage of image noises, we carefully devise three novel modules. The first is the *multi-scale high-frequency feature extraction module*. We adopt the widely used high-pass filters from SRM [18] to extract high-frequency noises from images. Unlike [11, 52, 57] that only consider extracting noises from an input image, we further apply these filters to low-level features at multiple scales to compose more abundant and informative features. Employing both the high-frequency noises and the low-frequency textures, we build a two-stream network to process the two modalities, respectively. Secondly, we apply the *residual guided spatial attention* in the entry part to guide the RGB modality from the residual perspective to attach more importance to forgery traces. Thirdly, we design a *dual cross-modality attention module* to formulate the interaction between the two modalities instead of keeping them independent [11, 52]. In this way, the two modalities provide complementary information based on their correlation and mutually promote representation learning.

Our contributions are summarized as follows:

- We perform an analysis on CNN-based detectors and find that they are biased to method-specific textures, leading to the generalization problem. Given that the image's high-frequency noises can remove color textures and reveal forgery traces, we propose to utilize image noises to boost the generalization ability.
- We devise a generalizable model by exploiting high-frequency features and modeling the correlation and interaction between the high-frequency modality and the regular one. We design three functional modules for learning an adequate representation, *i.e.*, the multi-scale high-frequency feature extraction module, the residual guided spatial attention module, and the dual cross-modality attention module.
- We conduct comprehensive evaluations on several benchmarks and demonstrate the superior generalization ability of the proposed model.

2. Related Work

Conventional image forgery detection. Though image tampering detection has been investigated for a long time, those conventional detectors [16, 39] cannot handle the detection of face forgery well. Firstly, those methods detect image editing operations like copy-move or copy-paste [8], which produce different artifacts from those in GAN-based face forgeries. Besides, face forgeries have a much smaller size and inferior quality than natural images. Furthermore, recent advanced forgery techniques leave almost no visible artifact in the tampered face, which easily deceive conventional detectors and require specialized treatments.

Specific artifacts or novel architectures. Current attempts mainly focus on improving within-database performance. [29, 34, 55, 9, 7, 32] targeted at specific artifacts such as abnormal eye-blinking frequency [29] or head-pose inconsistency [55]. However, those artifacts may not exist in improved forgeries. For network design, Afchar *et al.* [6] provided two compact network architectures to capture the mesoscopic features. Nguyen *et al.* [36] introduced the capsule network. These methods emphasize the power of representation and computational efficiency but did not explicitly consider the generalization ability.

Auxiliary tasks or synthetic data. Some works notice the generalization problem and attempt to utilize additional tasks [14, 35, 45, 44] or generate samples capturing the typical defects [30, 28, 54]. Stehouwer *et al.* [45] proposed to jointly predict the binary label and the attention map of the manipulated region. Yang *et al.* [54] generated samples with similar distributions to those in the target domain. These methods require additional annotations or extra data samples. Li *et al.* [30] observed that forged images contain some common warping and blurring effects. Li *et al.* [28] focused on the fusion operation in forgery creation. Both of them customized a database to learn such artifacts, but there remain non-negligible gaps between the hand-crafted forgeries and those from sophisticated algorithms.

High-frequency features. Several methods tried to exploit information from other domains. Durall *et al.* [15] found that the spectrum of real and fake images distributes differently in the high-frequency part. Chen *et al.* [11] proposed a multi-stream design and leveraged the DFT features. Wu *et al.* [52] proposed a two-stream network to extract spatial features and steganalysis features, respectively. Qian *et al.* [38] applied DCT on images and collected the frequency-aware clues to mine subtle forgery artifacts and compression errors. These methods exploited frequency information but did not explicitly consider the relationship between different domains.

3. Analysis of Generalizable Forgery Detector

3.1. Why current methods fail to generalize?

CNN and texture bias. Although many previous methods perform flawlessly on the testing set, they always suffer from a significant performance drop on images manipulated by unseen algorithms (See Tab. 1). Why do they fail to generalize? The reason is that those deep CNN models learn to capture the method-specific texture patterns for forgery

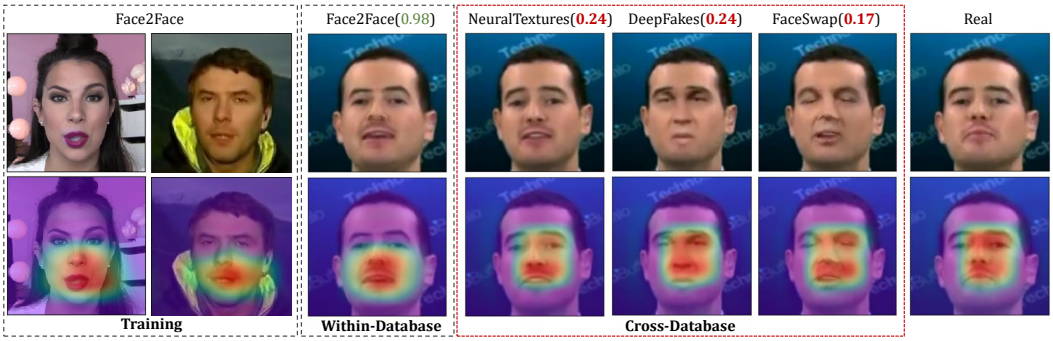

Figure 2: Grad-CAM maps from the Xception model trained on F2F forgeries. Numbers in the bracket denote the probability of being classified as fake. The mouth region is especially highlighted in F2F images, indicating that the model learns F2F's specific texture. But when evaluated on unseen forgeries (*i.e.,* NT, DF, and FS), the model's responses resemble those in the real face more, showing that it fails to recognize identical artifacts and mistakes these forgeries as real.

Table 1: Xception [12] detector experiences a significant performance drop when evaluated on unseen forgeries.

Training Set	Testing Set (AUC)			
	DF	F2F	FS	NT
DF	0.993	0.736	0.485	0.736
F2F	0.803	0.994	0.762	0.696
FS	0.664	0.888	0.994	0.713
NT	0.800	0.813	0.731	0.991

detection [32]. Geirhos *et al.* [19] studied the texture response of CNNs and showed that CNN models are strongly biased to textures. Different forgery algorithms always have unique network architectures and processing streams, so images manipulated by different algorithms will have different *fake textures*. Therefore, it is hard for a CNN model that has already been biased to one kind of fake textures to generalize to another.

Grad-CAM study and results. To verify that the CNN classifier is biased to specific fake textures, we exploit the gradient-based visualization tool [42] to generate class activation maps, namely Grad-CAM maps, which reveal the regions that CNNs rely on for classification. We train Xception [12] on images forged by the Face2Face method and then evaluate it on forgeries of four different algorithms.

As presented in Fig. 2, in the within-database evaluation, the model correctly finds out forgeries with high confidence. It especially concentrates on regions around the mouth (warmer color region), just as it does on the training images. Nevertheless, when encountering unseen forgeries, the model mistakes the other three kinds of fakes as real though they contain many visible artifacts. Besides, the high response regions deviate significantly from those in the training set and resemble those in the real face. The reason is that the model has overfitted to F2F's unique fake textures. Thus, when it cannot recognize the identical fake texture patterns, it always gives wrong predictions.

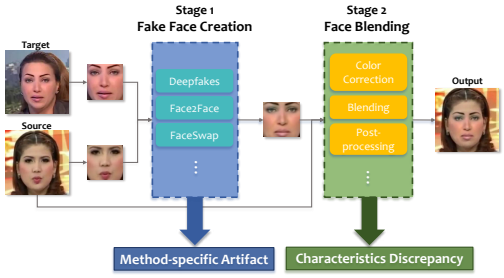

Figure 3: The overview of the typical manipulation pipeline. The two stages bring the method-specific texture artifacts and characteristics discrepancy, respectively.

The above observation motivates us to investigate more general clues other than the texture patterns.

3.2. What is common in forged face images?

Manipulation pipeline. To find the commonality among different manipulation methods, let us review the typical face manipulation pipeline firstly. As presented in Fig. 3, the manipulation procedure can be roughly divided into two processing stages, *i.e.*, *fake face creation* and *face blending*. In the first stage, fake faces are generated by sophisticated deep networks [46, 3] or rendered based on model templates [47, 5]. Different manipulation methods always adopt different algorithms and thus produce varied texture patterns. In the second stage, the generated face is further refined and warped back to the original image. This stage usually consists of some post-processing operations such as color correction and image blending.

Discrepancy brought by forged part. Though the face region in the output image is manipulated, the background remains the same as in the source image (See Fig. 3). Assuming that different images have unique characteristics, the blending stage violates the original data distributions, and we can utilize the characteristics discrepancy to gener-

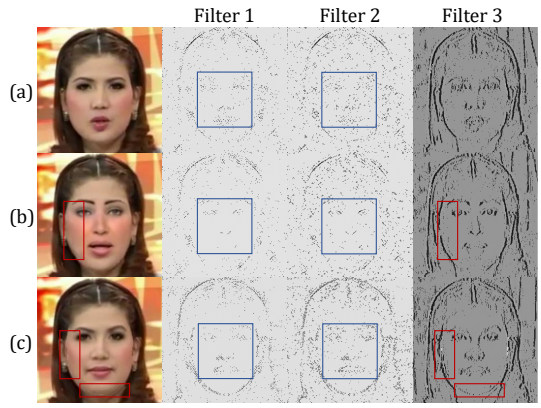

Figure 4: Noises extracted by SRM from (a) authentic face and face manipulated by (b) Deepfakes and (c) Face2Face. Red boxes mark blending traces that are hard to recognize in the RGB space but distinctive in the noise space. Blue boxes mark noise statistics in the central face region. Noises in the face region distribute continuously in authentic images but are visibly smoother or sharper in tampered images.

alize forgery detection. We share a similar intuition to [28], which focuses on the blending operation and detects the blending boundary. But instead of identifying the *boundary pattern* which may change with various post-processing operations, we expect a more robust way to discover the inconsistency between authentic and tampered regions.

3.3. Use SRM to extract high-frequency noise features and boost the generalization ability

Based on the above analysis, we assume that a generalizable forgery detector should (i) pay attention not only to texture-related features but also to texture-irrelevant ones, and (ii) be capable of discovering the discrepancy between the tampered face and pristine background. Observing that the image's high-frequency noises remove the color content while portrait the intrinsic characteristics, we attempt to utilize image noises for face forgery detection.

Image Noise Analysis. Noises are some high-frequency signals capturing random variations of brightness or color information. The distributions of image noises are influenced by the image sensor or circuitry of a digital camera. Hence images processed by different equipment or coming from different sources have different noise patterns. Noises can be viewed as an intrinsic specificity of an image and can be found in various forms in all digital imagery domains [25]. Given that manipulation operations ruin the consistency of characteristics in the original image, there often leave distinctive traces in the noise space [17, 33].

SRM for noise feature extraction. Inspired by recent progress on SRM [18] noise features in general image manipulation detection [58, 53], we adopt SRM filters for noise extraction. Fig. 4 shows some examples of SRM noises.

To validate noise features' effectiveness, we compare the detection performance on RGB color textures and SRM noises features, respectively. We train two Xception models, one with the regular RGB images and the other with the SRM noises. The two models are trained on images forged by the F2F method and evaluated on all four methods. As presented in the first two rows in Tab. 3, the model with SRM noise generalizes better than that with the regular color textures, especially on FS and NT methods.

4. The Proposed Method

The promising performance of SRM noises motivates us to explore the noise space further and boost generalization. In this section, we devise three modules to take full advantage of the high-frequency features, *i.e.,* a multi-scale high-frequency feature extraction module, a dual cross-modality attention module, and a residual guided spatial attention module. The proposed model is illustrated in Fig. 5.

4.1. Multi-scale High-frequency Feature Extraction

Previous methods [57, 53, 11] extract noise residuals solely from the input image. Apart from a straight-forward conversion of the input image, we apply high-pass filters to multiple low-level feature maps to enrich the high-frequency features. As shown in Fig. 5, given an input RGB image $\mathbf{X}$, we convert it to a residual image $\mathbf{X}_h$ in the high-frequency domain exploiting the SRM filters [18]. The entry flow takes both $\mathbf{X}$ and $\mathbf{X}_h$ as input and generates two types of raw features, *i.e.,* the multi-scale high-frequency feature maps $\mathbf{F}_h$ and the low-frequency spatial feature maps $\mathbf{F}$, which can be formulated as

$$\mathbf{F}, \mathbf{F}_h = f_{\text{entry}}(\mathbf{X}, \mathbf{X}_h). \tag{1}$$

The multi-scale high-frequency features are obtained as follows. Firstly, we apply regular convolutions on $\mathbf{X}$ and $\mathbf{X}_h$ to produce feature maps $\mathbf{F}^1$ and $\mathbf{F}_h^1$. To extract more high-frequency information, we then apply SRM filters on $\mathbf{F}^1$, following a 1×1 convolution to align the channel dimensions and get the output $\tilde{\mathbf{F}}_h^1$. As $\mathbf{F}_h^1$ and $\tilde{\mathbf{F}}_h^1$ are obtained from different sources and different operations, they contain different information. We down-sample $\mathbf{F}_1$ as well as the sum of $\mathbf{F}_h^1$ and $\tilde{\mathbf{F}}_h^1$, and then get $\mathbf{F}^2$ and $\mathbf{F}_h^2$ in the two streams, respectively. Repeating the above operations, we finally acquire the multi-scale high-frequency feature maps $\mathbf{F}_h$. Compared with $\mathbf{X}_h$, $\mathbf{F}_h$ embodies more abundant high-frequency signals from both the image and low-level feature maps at multiple scales.

4.2. Residual Guided Spatial Attention

Inspired by CBAM [51], we adopt spatial attention to highlight the manipulation traces and guide the feature

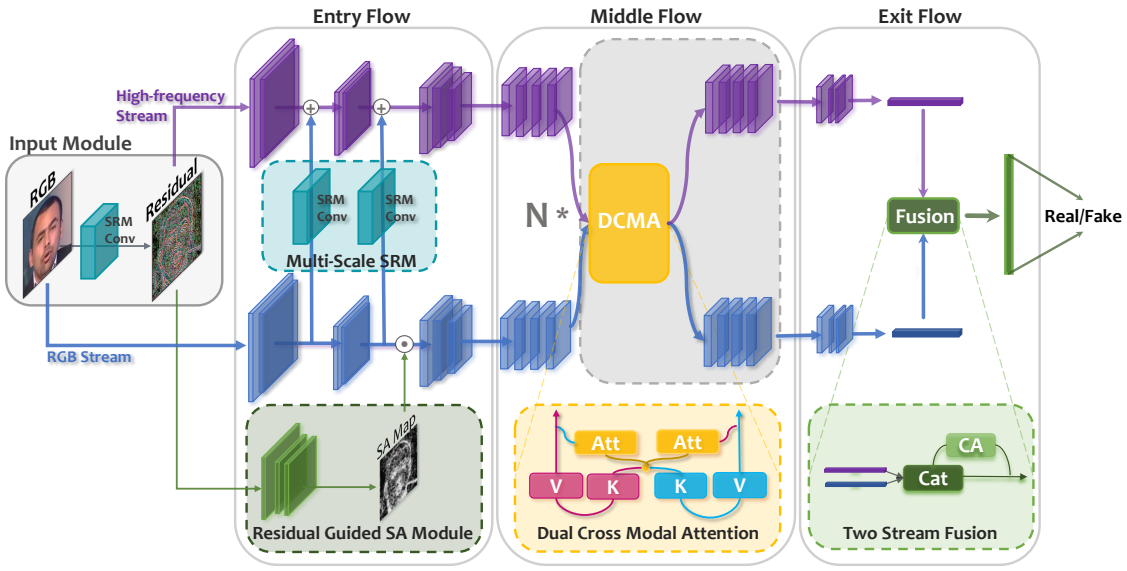

Figure 5: The pipeline of the proposed model. We design a two-stream architecture to process the RGB image and the high-frequency noises. In the entry flow, we extract multi-scale high-frequency features and residual guided spatial features. In the middle flow, we model the interaction between feature maps of the two modalities via several DCMA modules. Features from the two streams are fused in an attention-based manner for the final classification.

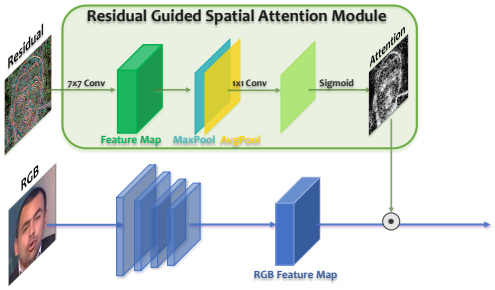

Figure 6: The residual guided spatial attention.

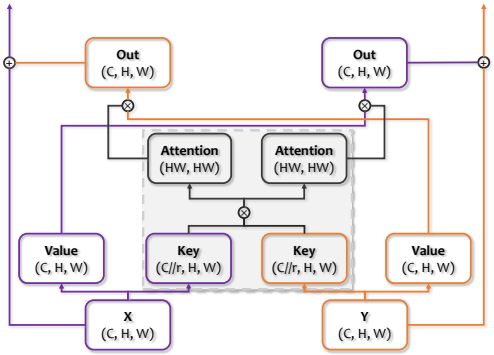

Figure 7: The dual cross-modality attention.

learning in the RGB modality. While CBAM derives the attention weights from the RGB image, we exploit the residuals generated by the SRM filters to predict the attention maps, which guide from a high-frequency perspective (See Fig. 6). Considering that the spatial correspondence between the noise residuals and low-level RGB features is still well maintained, we place several spatial attention modules in the entry part. The residual guided spatial attention block is defined as

$$\mathbf{M} = f_{\text{att}}(\mathbf{X}_h), \tag{2}$$

where $\mathbf{M}$ is the output attention map and $\mathbf{X}_h$ is the residual image. The raw feature maps $\mathbf{F}$ in Eq. (1) are computed by feeding the element-wise production $\mathbf{M} \odot \mathbf{F}^2$ into consequent convolutions and down-sampling operations.

See the attention visualizations in Fig. 8, high responses occur around abnormal facial boundaries in manipulated faces but distribute uniformly in real ones, which implies that the residual guided spatial attention can help feature extractor focus on forgery traces.

4.3. Dual Cross-modality Attention

The attention mechanism has been broadly applied in natural language processing [48] and computer vision [22, 37, 23]. As for the cross-modality attention, Ye *et al.* [56] applied self-attention to concatenated features, while Hou *et al.* [21] derives correlation-based attention for paired class and query features. Inspired by these works, we devise a dual cross-modality attention module (DCMA) to capture long-range dependency and model the interaction between the low-frequency textures and the high-frequency noises.

Denoting input features derived from the RGB stream and the high-frequency stream as $\mathbf{T}$ and $\mathbf{T}_h$, respectively,

the DCMA module leverages the computation block described in Fig. 7 to convert them to $\mathbf{T}'$ and $\mathbf{T}'_h$:

$$\mathbf{T}', \mathbf{T}'_h = f_{\text{DCMA}}(\mathbf{T}, \mathbf{T}_h). \tag{3}$$

Taking the conversion of $\mathbf{T}$ as an example, we first convert the input $\mathbf{T} \in \mathbb{R}^{C\times H\times W}$ into two components through different convolution blocks. One value component $\mathbf{V} \in \mathbb{R}^{C\times H\times W}$ represents the domain-specific information, the other key component $\mathbf{K} \in \mathbb{R}^{C/r\times H\times W}$ measures the correlation between two different domains. r is a scalar that reduces $\mathbf{K}$'s channel dimension for computation efficiency. Input features $\mathbf{T}_h \in \mathbb{R}^{C\times H\times W}$ in the high-frequency domain are converted into two components, *i.e.,* $\mathbf{V}_h$ and $\mathbf{K}_h$, similarly. Note that the key component is obtained through a two-layer convolution block. We set the first layer independently while sharing the second layer to project feature maps from the two modalities onto the same space.

Then we measure the correlation $\mathbf{C} \in \mathcal{R}^{HW\times HW}$ between the two modalities through $\mathbf{C} = \text{flt}(\mathbf{K})^T \otimes \text{flt}(\mathbf{K}_h)$, where $\text{flt}(\cdot)$ denotes the flatten operation and $\otimes$ denotes matrix multiplication. For the RGB attention, we multiply the correlation $\mathbf{C}$ with weight matrix $\mathbf{W}$ and generate the attention map $\mathbf{A}$ through the softmax operation, which is formulated as $\mathbf{A} = \text{softmax}(\mathbf{C} \otimes \mathbf{W})$. Similarly, we obtain the high-frequency attention map $\mathbf{A}_h = \text{softmax}(\mathbf{C}^T \otimes \mathbf{W}_h)$. $\mathbf{A}$ and $\mathbf{A}_h$ re-weight features in one modality according to its correlation with the other.

Applying $\mathbf{A}$ to the corresponding value component $\mathbf{V}_h$, we obtain the refined features $\mathbf{R}$, where $\mathbf{R} = \text{flt}(\mathbf{V}_h) \otimes \mathbf{A}$. We then recover $\mathbf{R}$ to the input dimension, and output the features $\mathbf{T}'$ of the RGB stream by $\mathbf{T}' = \mathbf{T} + \mathbf{R}$. The calculation for features $\mathbf{T}'_h$ in the high-frequency stream is the same. $\mathbf{T}'$ and $\mathbf{T}'_h$ embody complementary information and promote the feature learning for each other. As shown in the pipeline, the DCMA module can be applied N times to model the correlations between the two modalities at different scales. We set $N = 2$ in our experiments.

Feature Fusion and Loss Function. High-level features of the two modalities are fused at the end of the exit flow (See Fig. 5). We apply channel-wise attention [22] on the concatenated features and then make the prediction. Inspired by progress in face recognition [50, 43], we adopt the AM-Softmax Loss [49] as the objective function since it leads to smaller intra-class variations and larger inter-class differences than the regular cross-entropy loss.

5. Experiments

5.1. Settings

Datasets. To evaluate the generalization ability, we perform experiments on five large scale benchmark databases, *i.e.,* FaceForensics++ (FF++) [40], DeepfakeDetection (DFD) [2], Deepfake Detection Challenge (DFDC) [1], CelebDF [31], and DeeperForensics-1.0 (DF1.0) [24]. Detailed specifications are presented in Tab. 2. For evaluation on FF++, we follow the official splits by using 740 videos for training, 140 videos for validation, and 140 videos for testing. There are three versions of FF++ in terms of compression level, *i.e.*, raw, lightly compressed (HQ), and heavily compressed (LQ). The heavier the compression level, the harder it to distinguish the forgery traces. Since realistic forgeries always have a limited quality, we use the HQ and LQ versions in experiments. **We adopt the HQ version by default and specify the version otherwise**.

Table 2: Specifications of benchmark databases.

Database	Video Scale	Manipulation Algorithm
FF++ [40]	1000 real, 4000 fake	DF [3], FS [5], NT [46], F2F [47]
DFD [2]	363 real, 3068 fake	Improved DF
CelebDF [31]	408 real, 795 fake	Improved DF
DFDC [1]	1133 real, 4080 fake	Unpublished
DF1.0 [24]	11,000 fake	DF-VAE [24]

Table 3: Ablation study on FF++. The metric is AUC. Results in gray indicate the within-dataset performance.

Method	DF	F2F	FS	NT
RGB	0.803	0.994	0.762	0.696
SRM	0.758	0.994	0.913	0.858
Two-stream Fusion (Fusion)	0.810	0.994	0.922	0.894
Fusion + RSA	0.819	0.995	0.966	0.927
Fusion + RSA + DCMA	0.801	0.995	0.975	0.957
Fusion + RSA + DCMA + Multi-scale	**0.837**	0.994	**0.987**	**0.984**

Implementation. We modify Xception [12, 40] as the backbone network. We use DLIB [41] for face extraction and alignment and resize the aligned faces to 256×256. Model parameters are initialized by Xception pre-trained on ImageNet. The batch size is set to 32. Adam [26] is used for optimization with a learning rate of 0.0002. Details of the two-stream network structure and training settings are presented in the supplementary material.

5.2. Ablation Study

To demonstrate the benefit of each module, we evaluate the proposed model and its variants on the FF++ database. All models are trained on F2F and examined on all four datasets. The results are presented in Tab. 3. RGB represents the Xception baseline with RGB images as input. SRM denotes the modified model with the input image replaced by SRM noise residuals. Two-stream Fusion represents the basic two-stream model, which consists of an RGB stream and a high-frequency noise stream and adopts the attention-based fusion mechanism. RSA, DCMA, and Multi-scale represent the residual guided spatial attention module, the dual cross-modality attention module, and the multi-scale high-frequency feature extraction module.

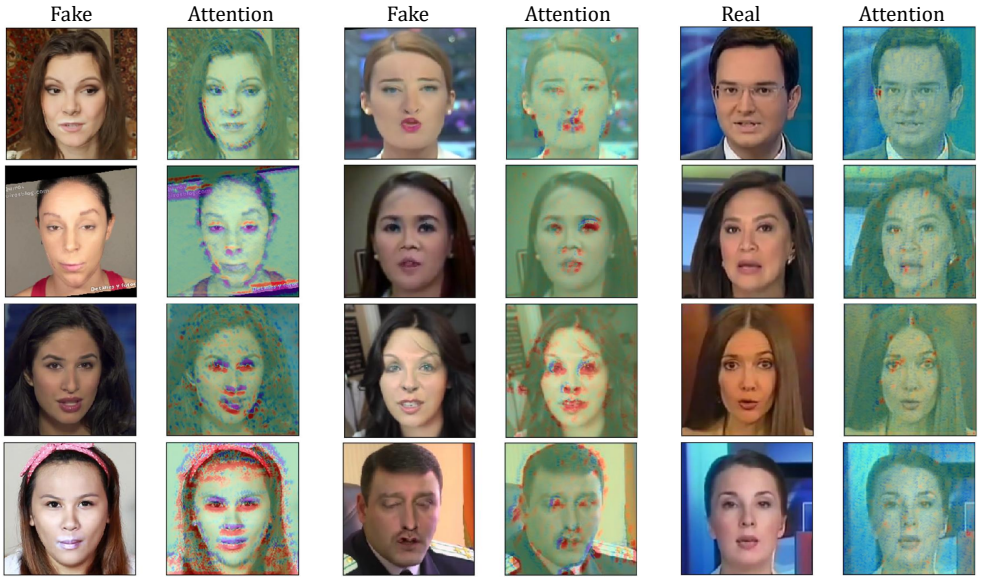

Figure 8: Examples of the residual guided attention. High responses occur around abnormal facial boundaries in manipulated faces while distribute uniformly in real ones.

Table 4: Cross-database evaluation on FF++ database (HQ).

Training Set	Model	Testing Set (AUC)			
		DF	F2F	FS	NT
DF	Xception [40]	**0.993**	0.736	0.490	0.736
	Face X-ray [28]	0.987	0.633	**0.600**	0.698
	Ours	0.992	**0.764**	0.497	**0.814**
F2F	Xception [40]	0.803	**0.994**	0.762	0.696
	Face X-ray [28]	0.630	0.984	0.938	0.945
	Ours	**0.837**	**0.994**	**0.987**	**0.984**
FS	Xception [40]	0.664	0.888	0.994	0.713
	Face X-ray [28]	0.458	0.961	0.981	0.957
	Ours	**0.685**	**0.993**	**0.995**	**0.980**
NT	Xception [40]	0.799	0.813	0.731	0.991
	Face X-ray [28]	0.705	0.917	0.910	0.925
	Ours	**0.894**	**0.995**	**0.993**	**0.994**

Table 5: Cross-database evaluation from FF++ to others.

Training	Model	Testing AUC			
		DFD	DFDC	CelebDF	DF1.0
FF++	Xception [40]	0.831	0.679	0.594	0.698
	Face X-ray [28]	0.856	0.700	0.742	0.723
	Ours	**0.919**	**0.797**	**0.794**	**0.738**

From this experiment we get the following observations. Firstly, compared with RGB, SRM achieves better performance on FS and NT. Furthermore, the basic two-stream model outperforms both RGB and SRM on all datasets, indicating the two modalities' complementary nature. As presented in the last three rows, the model's performance gradually improves with each module added step-by-step, demonstrating each module's effectiveness.

5.3. Generalization Ability Evaluation

To fully evaluate the proposed model's generalization ability, we conduct extensive cross-database evaluations in two different settings. The model is compared against two competing methods, Xception [40] and Face X-ray [28]. Face X-ray attempts to detect the fusion boundary brought by the blending operation. We implement it rigorously following the companion paper's instructions and train these models under the same setting.

Generalize from one method to another. We conduct this experiment on the FF++ (HQ) database [40] that contains forged images from four different manipulation techniques, *i.e.,* DeepFakes (DF) [3], Face2Face (F2F) [47], FaceSwap (FS) [5], and NeuralTextures (NT) [46]. We use forged images of one method for training and those of all four methods for testing. As shown in Tab. 4, our model exceeds the competitors in most cases. Since Xception overly relies on the texture patterns, its performance drops drastically in unseen forgeries. Face X-ray achieves a relatively better generalization ability as it detects the blending evidence. Our model leverages both textures and noises and captures the blending effects in the noise space, therefore generalizing better from one method to another.

Generalize from FF++ to other databases. In this experiment, we train models on FF++ (HQ) [40] and evaluate them in DFD [2], DFDC [1], CelebDF [31], and DF1.0 [24], respectively. This setting is more challenging than the first one since the testing sets share much less similarity with the training set. We can see from Tab. 5 that our method achieves apparent improvements over Xception and Face

X-ray on all the databases, especially on DFD, DFDC, and CelebDF. This is because Face X-ray learns to identify the boundary patterns that are sensitive to the post-process operations varying in different databases. On the contrary, our model learns more robust representations.

We present more statistics on the cross-database evaluation and the comparison result with Face X-ray on the blend face (BI) dataset in the supplementary material.

5.4. Comparison with recent works

Comparison with methods utilizing high-frequency features. We first compare with methods exploiting high-frequency features, *i.e.*, SRMNet [58], Bayar Conv [10], SSTNet [52], and F3Net [38]. The former three methods target video forges and share the same backbone but different high-pass filters, *i.e.*, SRM filters, Bayar Conv filters, and loosely-constrained residual filters. F3Net adopts Discrete Cosine Transform to estimate frequency statistics. We follow their setting and perform video-level detection on the highly compressed (LQ) FF++ database. Since our model is trained at the image level, we sample 1 frame every five frames to collect a total of 25 images for each video. Then we average the predictions over sampled images to classify each video. Results are presented in Tab. 6. Note that we compare against F3Net with the Xception backbone for a fair comparison. Our method achieves comparable robustness with F3Net and better performance than the others.

Comparison with methods using multi-task learning. We then compare our model against several methods that adopt multi-task learning for a better generalization ability, including LAE [14], ClassNSeg [35], and ForensicTrans [13]. These three methods perform forgery localization and classification simultaneously. Note that ForensicTrans adopts image residuals as input and needs to be fine-tuned on a few samples from the target domain. Following these methods, we train our model on F2F and test it on both F2F and FS. The results of the competing methods are the reported statistics in the corresponding papers. As illustrated in Tab. 7, our method surpasses the competitors in both within-database and cross-database evaluations.

Comparison with other state-of-the-art methods. We further compare our model with FWA [30] and FFD [45]. FWA focuses on post-processing artifacts such as blurring and warping effects, and it is trained on synthetic data which mimics those artifacts explicitly. FFD exploits annotated forgery masks to provide supervision on attention maps. Note that FFD is trained on the DFFD [45] database, which contains FF++ and forges from other manipulation algorithms. We adopt CelebDF as the testing set. As shown in Tab. 8, our method outperforms the competing methods by more than 15% in AUC.

Table 6: Comparison on FF++ with methods using high-frequency features. The metric is accuracy.

Model	Training/Testing Set (LQ)			
	DF	F2F	FS	Real
SRMNet [58]	0.919	0.927	0.891	0.693
Bayar Conv [10]	0.929	0.946	0.897	0.755
SSTNet [52]	0.934	0.919	0.919	0.793
F3Net [38]	0.980	0.953	**0.965**	-
Ours	**0.986**	**0.957**	0.929	**0.971**

Table 7: Comparison with recent works using multi-task learning. The metric is accuracy.

Training Set	Model	Testing Set (Acc)	
		FS(HQ)	F2F(HQ)
F2F(HQ)	LAE [14]	0.632	0.909
	ClassNSeg [35]	0.541	0.928
	ForensicTrans [13]	0.726	0.945
	Ours	**0.867**	**0.992**

Table 8: Comparison on CelebDF. The metric is AUC.

Model	Training Set	Testing AUC on CelebDF
FWA [30]	self-collected	0.538
FFD [45]	DFFD [45]	0.644
Ours	FF++(HQ)	**0.794**

6. Conclusion

In this paper, we conducted studies on the CNN-based forgery detector, finding that the CNN detector is easily overfitted to method-specific texture patterns. To learn more robust representations, we proposed to utilize the image's high-frequency noise features, which remove the color textures and reveal forgery traces. We leveraged both the color textures and the high-frequency noises and proposed a new face forgery detector. Three functional modules were carefully devised, *i.e.,* a multi-scale high-frequency feature extraction module, a residual guided spatial attention module, and a dual cross-modality attention module. We employed the proposed modules to extract more informative features and capture the correlation and interaction between complementary modalities. The comprehensive experiments demonstrate the effectiveness of each module, and the comparisons with the competing methods further corroborate our model's superior generalization ability.

Acknowledgement

This work was partly supported by the National Key Research and Development Program of China (2018AAA0100704), NSFC (61972250, U19B2035), CCF-Tencent Open Fund (RAGR20200113), and Shanghai Municipal Science and Technology Major Project (2021SHZDZX0102).

References

[1] Deepfake detection challenge. https://www.kaggle.com/c/deepfake-detection-challenge. Accessed: 2020-05-10. 6, 7

[2] Deepfakedetection. https://ai.googleblog.com/2019/09/contributing-data-to-deepfake-detection.html. Accessed: 2020-05-10. 6, 7

[3] Deepfakes. https://github.com/iperov/DeepFaceLab. Accessed: 2020-05-10. 3, 6, 7

[4] Faceapp. https://www.faceapp.com/. Accessed: 2020-05-10. 1

[5] Faceswap. https://github.com/MarekKowalski/FaceSwap. Accessed: 2020-05-10. 3, 6, 7

[6] Darius Afchar, Vincent Nozick, Junichi Yamagishi, and Isao Echizen. Mesonet: a compact facial video forgery detection network. In *WIFS*, 2018. 2

[7] Shruti Agarwal, Hany Farid, Yuming Gu, Mingming He, Koki Nagano, and Hao Li. Protecting world leaders against deep fakes. In *CVPRW*, 2019. 1, 2

[8] Irene Amerini, Lamberto Ballan, Roberto Caldelli, Alberto Del Bimbo, and Giuseppe Serra. A sift-based forensic method for copy–move attack detection and transformation recovery. *TIFS*, 2011. 2

[9] Irene Amerini, Leonardo Galteri, Roberto Caldelli, and Alberto Del Bimbo. Deepfake video detection through optical flow based cnn. In *ICCVW*, 2019. 1, 2

[10] Belhassen Bayar and Matthew C Stamm. 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*, pages 5–10, 2016. 8

[11] Zehao Chen and Hua Yang. Manipulated face detector: Joint spatial and frequency domain attention network. *arXiv preprint arXiv:2005.02958*, 2020. 1, 2, 4

[12] F. Chollet. Xception: Deep learning with depthwise separable convolutions. In *CVPR*, 2017. 3, 6

[13] Davide Cozzolino, Justus Thies, Andreas Rössler, Christian Riess, Matthias Nießner, and Luisa Verdoliva. Forensictransfer: Weakly-supervised domain adaptation for forgery detection. *arXiv preprint arXiv:1812.02510*, 2018. 1, 8

[14] Mengnan Du, Shiva Pentyala, Yuening Li, and Xia Hu. Towards generalizable forgery detection with locality-aware autoencoder. *arXiv preprint arXiv:1909.05999*, 2019. 1, 2, 8

[15] Ricard Durall, Margret Keuper, Franz-Josef Pfreundt, and Janis Keuper. Unmasking deepfakes with simple features. *arXiv preprint arXiv:1911.00686*, 2019. 2

[16] Hany Farid. Image forgery detection. *IEEE Signal processing magazine*, 2009. 2

[17] J. Fridrich. Digital image forensics. *IEEE Signal Processing Magazine*, 26(2):26–37, 2009. 4

[18] Jessica Fridrich and Jan Kodovsky. Rich models for steganalysis of digital images. *TIFS*, 2012. 2, 4

[19] Robert Geirhos, Patricia Rubisch, Claudio Michaelis, Matthias Bethge, Felix A. Wichmann, and Wieland Brendel. Imagenet-trained CNNs are biased towards texture; increasing shape bias improves accuracy and robustness. In *International Conference on Learning Representations*, 2019. 3

[20] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *NIPS*, 2014. 1

[21] Ruibing Hou, Hong Chang, MA Bingpeng, Shiguang Shan, and Xilin Chen. Cross attention network for few-shot classification. In *NIPS*, 2019. 5

[22] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *CVPR*, 2018. 5, 6

[23] Zilong Huang, Xinggang Wang, Lichao Huang, Chang Huang, Yunchao Wei, and Wenyu Liu. Ccnet: Criss-cross attention for semantic segmentation. In *ICCV*, October 2019. 5

[24] Liming Jiang, Ren Li, Wayne Wu, Chen Qian, and Chen Change Loy. Deeperforensics-1.0: A large-scale dataset for real-world face forgery detection. In *CVPR*, 2020. 1, 6, 7

[25] Thibaut Julliand, Vincent Nozick, and Hugues Talbot. Image noise and digital image forensics. volume 9569, pages 3–17, 03 2016. 4

[26] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *ICLR*, 2015. 6

[27] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *ICLR*, 2014. 1

[28] Lingzhi Li, Jianmin Bao, Ting Zhang, Hao Yang, Dong Chen, Fang Wen, and Baining Guo. Face x-ray for more general face forgery detection. *CVPR*, 2020. 1, 2, 4, 7

[29] Yuezun Li, Ming-Ching Chang, and Siwei Lyu. In ictu oculi: Exposing ai created fake videos by detecting eye blinking. In *WIFS*, 2018. 1, 2

[30] Yuezun Li and Siwei Lyu. Exposing deepfake videos by detecting face warping artifacts. *CVPRW*, 2019. 1, 2, 8

[31] Yuezun Li, Xin Yang, Pu Sun, Honggang Qi, and Siwei Lyu. Celeb-df: A new dataset for deepfake forensics. *arXiv preprint arXiv:1909.12962*, 2019. 6, 7

[32] Z. Liu, X. Qi, and P. H. S. Torr. Global texture enhancement for fake face detection in the wild. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8057–8066, 2020. 2, 3

[33] Babak Mahdian and Stanislav Saic. Using noise inconsistencies for blind image forensics. *Image and Vision Computing*, 27(10):1497 – 1503, 2009. Special Section: Computer Vision Methods for Ambient Intelligence. 4

[34] Falko Matern, Christian Riess, and Marc Stamminger. Exploiting visual artifacts to expose deepfakes and face manipulations. In *WACVW*, 2019. 2

[35] Huy H Nguyen, Fuming Fang, Junichi Yamagishi, and Isao Echizen. Multi-task learning for detecting and segmenting manipulated facial images and videos. *arXiv preprint arXiv:1906.06876*, 2019. 1, 2, 8

[36] Huy H Nguyen, Junichi Yamagishi, and Isao Echizen. Capsule-forensics: Using capsule networks to detect forged images and videos. In *ICASSP*, 2019. 2

[37] Marco Pedersoli, Thomas Lucas, Cordelia Schmid, and Jakob Verbeek. Areas of attention for image captioning. In *ICCV*, 2017. 5

[38] Yuyang Qian, Guojun Yin, Lu Sheng, Zixuan Chen, and Jing Shao. Thinking in frequency: Face forgery detection by mining frequency-aware clues. In Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm, editors, *Computer Vision – ECCV 2020*, pages 86–103, Cham, 2020. Springer International Publishing. 1, 2, 8

[39] Anderson Rocha, Walter Scheirer, Terrance Boult, and Siome Goldenstein. Vision of the unseen: Current trends and challenges in digital image and video forensics. *ACM Computing Surveys (CSUR)*, 2011. 2

[40] Andreas Rossler, Davide Cozzolino, Luisa Verdoliva, Christian Riess, Justus Thies, and Matthias Nießner. Faceforensics++: Learning to detect manipulated facial images. In *ICCV*, 2019. 6, 7

[41] Christos Sagonas, Epameinondas Antonakos, Georgios Tzimiropoulos, Stefanos Zafeiriou, and Maja Pantic. 300 faces in-the-wild challenge: Database and results. *IVC*, 2016. 6

[42] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 618–626, 2017. 3

[43] L. Song, D. Gong, Z. Li, C. Liu, and W. Liu. Occlusion robust face recognition based on mask learning with pairwise differential siamese network. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 773–782, 2019. 6

[44] Kritaphat Songsri-in and Stefanos Zafeiriou. Complement face forensic detection and localization with faciallandmarks. *arXiv preprint arXiv:1910.05455*, 2019. 2

[45] Joel Stehouwer, Hao Dang, Feng Liu, Xiaoming Liu, and Anil Jain. On the detection of digital face manipulation. *CVPR*, 2020. 1, 2, 8

[46] Justus Thies, Michael Zollhöfer, and Matthias Nießner. Deferred neural rendering: Image synthesis using neural textures. *TOG*, 2019. 3, 6, 7

[47] Justus Thies, Michael Zollhofer, Marc Stamminger, Christian Theobalt, and Matthias Nießner. Face2face: Real-time face capture and reenactment of rgb videos. In *CVPR*, 2016. 3, 6, 7

[48] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NIPS*, 2017. 5

[49] Feng Wang, Jian Cheng, Weiyang Liu, and Haijun Liu. Additive margin softmax for face verification. *IEEE Signal Processing Letters*, 2018. 6

[50] H. Wang, Y. Wang, Z. Zhou, X. Ji, D. Gong, J. Zhou, Z. Li, and W. Liu. Cosface: Large margin cosine loss for deep face recognition. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5265–5274, 2018. 6

[51] Sanghyun Woo, Jongchan Park, Joon-Young Lee, and In So Kweon. Cbam: Convolutional block attention module. In *ECCV*, 2018. 4

[52] Xi Wu, Zhen Xie, YuTao Gao, and Yu Xiao. Sstnet: Detecting manipulated faces through spatial, steganalysis and temporal features. In *ICASSP*, 2020. 1, 2, 8

[53] Yue Wu, Wael AbdAlmageed, and Premkumar Natarajan. Mantra-net: Manipulation tracing network for detection and localization of image forgeries with anomalous features. In *CVPR*, 2019. 4

[54] Chao Yang and Ser-Nam Lim. One-shot domain adaptation for face generation. *CVPR*, 2020. 1, 2

[55] Xin Yang, Yuezun Li, and Siwei Lyu. Exposing deep fakes using inconsistent head poses. In *ICASSP*, 2019. 2

[56] Linwei Ye, Mrigank Rochan, Zhi Liu, and Yang Wang. Cross-modal self-attention network for referring image segmentation. In *CVPR*, 2019. 5

[57] Peng Zhou, Xintong Han, Vlad I Morariu, and Larry S Davis. Two-stream neural networks for tampered face detection. In *CVPRW*, 2017. 1, 2, 4

[58] Peng Zhou, Xintong Han, Vlad I Morariu, and Larry S Davis. Learning rich features for image manipulation detection. In *CVPR*, 2018. 4, 8

[59] Hao Zhu, Chaoyou Fu, Qianyi Wu, W. Wu, Chen Qian, and R. He. Aot: Appearance optimal transport based identity swapping for forgery detection. *ArXiv*, abs/2011.02674, 2020. 1