Face Forgery Detection via Multi-Feature Fusion and Local Enhancement

Dengyong Zhang, Jiahao Chen, Xin Liao, Feng Li, Jiaxin Chen, and Gaobo Yang

***Abstract*—With the rapid growth of Internet technology, security concerns have risen, particularly with the prevalence of Deepfakes, a popular visual forgery technique. Therefore, there is necessary to research more powerful methods to detect Deepfakes. However, many Convolutional Neural Networks-based detection methods struggle with cross-database performance, often overfitting to specific color textures. We observe that image noises can weaken the influence of color textures and expose the forgery traces in the noise domain. This is because tampering techniques, when altering face images, disrupt the consistency of feature distribution in the noise space. And the forgery traces in the noise space are complementary to the tampering artifacts present in the image space information. Therefore, we propose a novel face forgery detection network that combines spatial domain and noise domain. Our Dual Feature Fusion Module and Local Enhancement Attention Module contribute to more comprehensive feature representations, enhancing our method's discriminative ability. Experimental results demonstrate superior performance compared to existing methods on mainstream datasets. https://github.com/jhchen1998/DeepfakeDetection.**

***Index Terms*—Deepfakes, Face Forgery Dection, Spatial domain, Noise domain.**

I. Introduction

DEEPFAKES is the combination of the words "Deep" and "Fakes", it has become an iconic word of face forgery technology. People can replace the face of the target character with the face of another person, in the corresponding position within a video, using Deepfakes technology to create a fake video with the target character's face. As Deep Generative Models show strong generative ability, they are gradually used in face tampering. In the current era of the internet, countless videos are uploaded daily, including many fake videos. Such a trend may cause several security issues and problems. Therefore, researchers active in the field of facial forgery have begun to explore methods of detecting DeepFake videos. Early Deepfake detection research mainly relied on traditional images and videos processing techniques, such as Discrete Fourier Transform (DFT), Local Binary Patterns (LBP), steganographic feature analysis. Although these methods can detect Easydeepfakes, as Deepfake technology advances, the traditional detection techniques lose efficiency and effectiveness, causing a shift towards deep learning-based detection methods.

Deep learning provides a promising pathway to address this challenge. Recently, face forgery detection methods [1]–[7] based on deep learning have made remarkable progress by mining forgery clues using Convolutional Neural Networks (CNNs) on mainstream datasets, such as the FaceForensics++ (FF++) dataset [8]. But face forgery detection methods based on deep learning struggle to achieve satisfactory performance under cross-datasets conditions due to overfitting to specific color textures. Nonetheless, researchers have proposed frequency-based detection methods, such as using Discrete Cosine Transform (DCT) and DFT, to detect forgery clues in the frequency domain space [9], [10].

Some previous articles [11], [12] considered Deepfake detection from the perspective of the noise domain because noise information can highlight more subtle forgery traces. These forgery clues in the noise space cannot be directly detected in the spatial domain. However, the consistency of features in the original image is destroyed during the fake image generation process, leaving unique traces in the noise space. By incorporating high-frequency noise, researchers can eliminate the original color texture of the image and reveal hidden forgery traces. Therefore, utilizing noise features for face forgery detection is an essential research direction.

In this paper, we propose a general face forgery detection method that enhances detection ability by combining high-frequency noises and spatial information. To better fuse these two types of information and make use of their complementary nature, we design a Dual Feature Fusion Module. This module combines high-frequency noise information obtained through Steganalysis Rich Model (SRM) with spatial texture information obtained by CNNs to gain a more informative feature representation. We also propose a Local Enhancement Attention Module to help the model focus on and highlight tampering traces in order to achieve better detection results.

II. Related Work

Historically, earlier methods of facial manipulation frequently lead to clearly discernible forged artifacts. This has consequently driven the creation of numerous detection methods aimed at identifying anomalies within the spatial domain [13]–[20]. Haliassos et al. [13] believed that fake videos have flaws in the coherence of lips. Jung et al. [14] proposed a method to detect fake videos based on human eye blinking pattern. Yu et al. [17] constructed a dual-stream prediction network to alleviate the issues of feature omission

This work was funded in part by the National Natural Science Foundation of China under Grant 62172059,62272160, U22A2030 and 61972142, in part by Scientific Research Fund of Hunan Provincial Education Department of China under Grant 22A0200. (Corresponding author: Xin Liao)

Dengyong Zhang, Jiahao Chen, Feng Li and Jiaxin Chen are with School of Computer and Communication Engineering, Changsha University of Science and Technology, Changsha 410004, China (e-mail: zhdy@csust.edu.cn; cjh_160188@stu.csust.edu.cn; lif@csust.edu.cn; chenjiaxin@hnu.edu.cn).

Xin Liao and Gaobo Yang are with Hunan University, Changsha 410082, China (e-mail: xinliao@hnu.edu.cn; yanggaobo@hnu.edu.cn).

This article has been accepted for publication in IEEE Transactions on Circuits and Systems for Video Technology. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2024.3390945

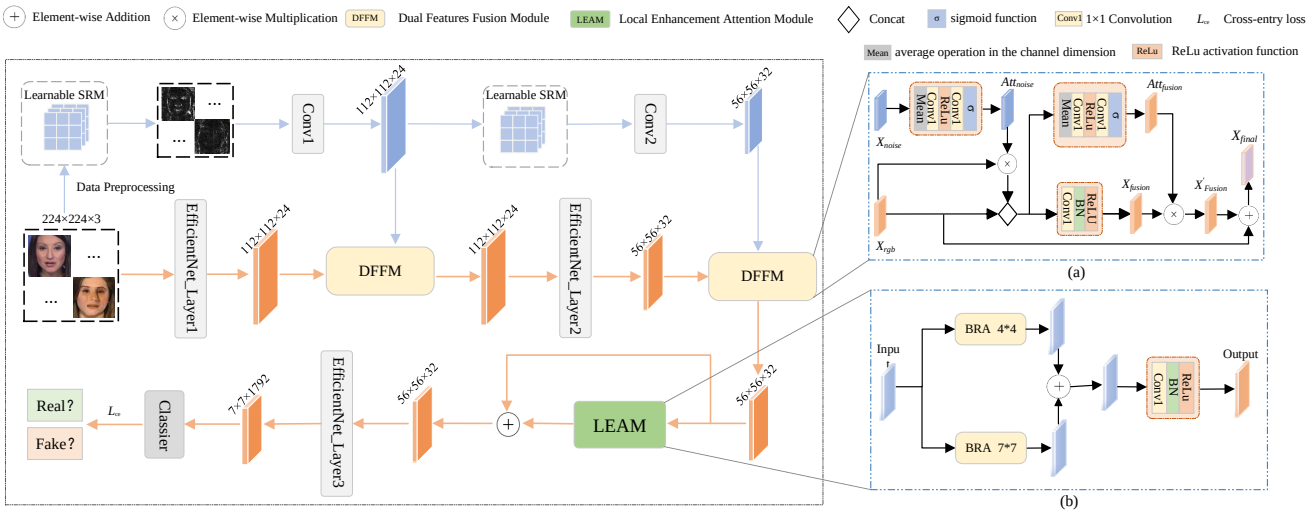

Fig. 1. Overview of the proposed method. Our proposed network structure utilizes a two-stream architecture that mines tampering traces in both the spatial and noise domain. Specifically, we use EfficientNet to extract spatial information, with EfficientNet_Layer1, EfficientNet_Layer2 and EfficientNet_Layer3 representing the three parts of EfficientNet. Learnable SRM filters are employed to learn noise information. Conv1 and Conv2 represent the convolution operation with a convolution kernel size of 3×3 and a stride of 2. Figure (a) represents Dual Features Fusion Module, and figure (b) represents Local Enhancement Attention Module.

and redundancy. Zhao et al. [18] extracted texture information through spectral analysis and integrated it with RGB features to enhance detection performance.

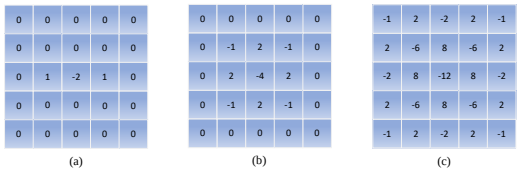

Fig. 2. Three SRM filters with fixed weights are utilized to initialize the parameters of the learnable SRM filters. Figure (a), (b) and (c) represent three filters with different parameter settings.

As tampering technology continues to evolve, these algorithms will gradually lose their detection ability as flaws are repaired. And some researchers have begun to apply various CNNs methods to min forged artifacts [1], [2], [15], such as Xception [1] and EfficientNet [2].

Researchers are not only leveraging spatial features but also integrating CNNs and Visual Transformers [21] with frequency domain features. Some researchers [9], [11], [19], [22], [23] have identified that tampering leaves certain features in the frequency domain, which cannot be observed directly in the RGB image but can be acquired through spectrum analysis. For example, Qian et al. [9] and Wu et al. [22] utilized DCT to extract frequency domain features for detection. Luo et al. [23] and Zhou et al. [11] observed that image noises can improve the detection ability for face forgery detection. And Liu et al. [19] proposed a network architecture that integrates local related features and frequency information to uncover forgery patterns, enhancing detection performance using the multi-feature enhancement module.

III. Our Approach

A. Overview

We propose a method that adopts a fusion approach to extract features by combining spatial information flow and noise information flow. The proposed network structure is shown in Fig. 1. We adopt EfficientNet [2] as the backbone network for extracting spatial features, and we use learnable SRM filters to extract noise features. Our learnable SRM filters are initialized with three fixed kernels, whose weights are shown in Fig. 2. To achieve better integration of high-frequency noise information with spatial information, we have designed the Dual Features Fusion Module (DFFM) to assist in the fusion of information. DFFM enables us to retain the original feature information while incorporating the high-frequency noise information we extracted into the features.

After fusing spatial and noise information, we enhance the resulting features by Local Enhancement Attention Module (LEAM). LEAM captures long-range dependencies using self-attention mechanisms to increase the expressive power of local features. In LEAM, we use two Bi-Level Routing Attention mechanisms to extract features, each with a different patch size. Extracting features at various scales increases the richness of the features. Afterwards, we further pass the enhanced features to the backbone and perform classification at the end.

B. Dual Feature Fusion Module

We propose the Dual Feature Fusion Module to fuse noise information and spatial information, thereby obtaining a more informative feature representation. The structure diagram is shown in Fig. 1 (a). DFFM takes two input features: RGB features X_{rgb} from the spatial stream and noise features X_{noise} from the noise stream. For noise features X_{noise}, we

TABLE I
THE EVALUATION OF VARIOUS DETECTION METHODS ON FACEFORENSICS++ AND FACESHIFTER DATASETS. BEST RESULTS ARE MARKED IN BOLD.

Methods	FF++ (HQ)		FF++ (LQ)		FaceShifter (HQ)		FaceShifter (LQ)	
	ACC	AUC	ACC	AUC	ACC	AUC	ACC	AUC
Meso-4 [15]	50.02%	63.72%	50.47%	62.70%	70.20%	88.52%	57.06%	73.42%
MesoInception-4 [15]	56.46%	88.90%	60.55%	73.74%	88.40%	99.07%	82.41%	95.44%
Xception [1]	94.04%	98.42%	76.35%	87.34%	96.64%	99.75%	93.56%	98.17%
GramNet(resnet18) [16]	91.91%	97.75%	72.77%	85.37%	97.51%	99.62%	92.80%	97.59%
GocNet [24]	92.17%	97.65%	75.77%	85.56%	97.67%	99.53%	90.70%	96.73%
RFM [25]	93.62%	98.53%	75.67%	87.18%	98.03%	99.70%	93.16%	98.46%
EfficientNet-B4 [2]	93.19%	98.77%	76.60%	89.09%	97.41%	99.76%	93.36%	**98.53%**
M2TR [21]	94.08%	98.43%	72.73%	89.23%	97.80%	99.65%	92.69%	98.38%
GFFD [23]	92.22%	98.44%	75.52%	86.36%	96.09%	99.70%	91.81%	98.23%
Ours	**94.14%**	**98.78%**	**79.58%**	**89.93%**	**98.67%**	**99.83%**	**94.07%**	98.48%

perform a series of operations to obtain the attention map Att_{noise}. These operations can be expressed as:

$$Att_{noise} = \sigma\left(Conv1\left(Re\left(Conv1\left(Mean\left(X_{noise}\right)\right)\right)\right)\right) \quad (1)$$

Where $Mean$ denotes a average operation in the channel dimension. $Conv1$ has a stride of 1 and a kernel size of 1. Re is $RELU$ activation function, σ is the sigmoid function. After obtaining the attention map provided by the noise information, we then multiply this attention map with the input of the spatial stream to obtain a new feature map $X^{'}_{rgb}$. This step can be expressed as:

$$X^{'}_{rgb} = X_{rgb} \bigodot Att_{noise} \quad (2)$$

Where $\bigodot$ represents the Hadamard product, and we splice the new feature map $X^{'}_{rgb}$ with the original feature map in the RGB stream according to channel dimension. Then, we use a 1×1 convolutional layer to obtain a feature X_{fusion} that combines RGB and noise information. This step can be expressed as:

$$X_{fusion} = Re\left(BN\left(Conv1\left(Cat\left(X_{rgb}, X^{'}_{rgb}\right)\right)\right)\right) \quad (3)$$

Where Cat represents the concat operation, and BN represents Batch Normalization. The main purpose of this step is to fuse the noise features with the RGB features, resulting in a fusion feature map that has a more enriched representation of features. After obtaining the fusion feature X_{fusion}, we perform channel attention on it to obtain a new feature map $X^{'}_{fusion}$. This operation can be expressed as:

$$Att_{fusion} = \sigma\left(Conv1\left(Re\left(Conv1\left(Mean\left(X_{fusion}\right)\right)\right)\right)\right) \quad (4)$$

$$X^{'}_{fusion} = X_{fusion} \bigodot Att_{fusion} \quad (5)$$

After performing the above series of operations, we obtain the feature map $X^{'}_{fusion}$ that combines the noise features with the Spatial features. We then add the $X^{'}_{fusion}$ to the original feature to get the final feature X_{final}. Finally, we send X_{final} to the backbone network for further processing, this step can be expressed as:

$$X_{final} = X_{rgb} + X^{'}_{fusion} \quad (6)$$

TABLE II
CROSS-DATASET EVALUATION RESULTS ON CELEB-DF, DFDC AND WILDDEEPFAKE DATASETS.

Methods	Training Set	Testing Set (AUC)		
		Celeb-DF	DFDC	WildDeepfake
Meso-4 [15]	FF++ (HQ)	58.31%	49.31%	68.51%
MesoInception-4 [15]	FF++ (HQ)	56.70%	54.32%	62.85%
Xception [1]	FF++ (HQ)	62.61%	66.84%	70.41%
GramNet(resnet18) [16]	FF++ (HQ)	66.31%	65.97%	70.05%
EfficientNet-B4 [2]	FF++ (HQ)	66.44%	66.60%	70.73%
RFM [25]	FF++ (HQ)	65.21%	67.33%	72.99%
MSFRNet [17]	FF++ (HQ)	69.84%	-	-
F3-Net [9]	FF++ (HQ)	65.17%	57.87%	60.49%
M2TR [21]	FF++ (HQ)	66.04%	69.00%	71.69%
Ours	FF++ (HQ)	**69.84%**	**71.28%**	**73.04%**

C. Local Enhancement Attention Module

In order to improve the expressive ability of local features, some scholars previously proposed the use of self-attention to capture long-range dependencies in images [26]. Based on these studies, we propose a Local Enhancement Attention Module (LEAM). This module enables improved representation of local features. We integrate two Bi-Level Routing Attention (BRA) [27] of different scales into LEAM. BRA is a sparse attention mechanism that utilizes a self-attention mechanism and dual routing mechanism for dynamically selecting the most relevant adjacent feature regions. Its structure is shown in Fig. 3, and the structure of LEAM is shown in Fig. 1 (b). The basic idea is to partition the input feature map into multiple regions and then enable each region to adaptively select the most relevant adjacent regions while filtering out the irrelevant regions.

Assuming we are given a feature map input $X \in \mathbb{R}^{H \times W \times C}$, we first divide it into $S \times S$ non-overlapping regions, each region contains $\frac{HW}{S^2}$ feature vectors. In this paper, we set the size of the non-overlapping region of two BRA mechanisms to 4×4 and 7×7, and S is $\frac{W}{4}$ and $\frac{W}{7}$ respectively.

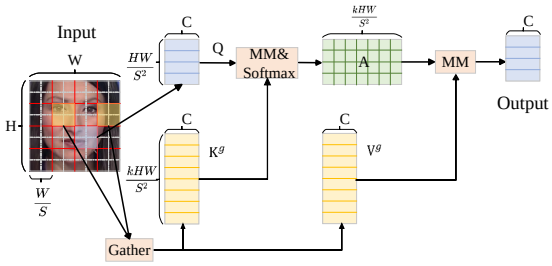

Fig. 3. The structure of Bi-Level Routing Attention. MM is the matrix multiplication function.

Thus, we obtain a reshaped feature map $X^r \in \mathbb{R}^{S^2 \times \frac{HW}{S^2} \times C}$. And then, we proceed to linearly map to obtain the required $Q, K, V \in \mathbb{R}^{S^2 \times \frac{HW}{S^2} \times C}$. Next, we obtain regional-level queries and keys $Q^r, K^r \in \mathbb{R}^{S^2 \times C}$, respectively by computing the regional-level averages of Q, K. We can construct the adjacency matrix $A^r \in \mathbb{R}^{S^2 \times S^2}$ of a region-to-region directed graph, by performing matrix multiplication between Q^r and the transpose of K^r:

$$A^r = Q^r \times (K^r)^T \tag{7}$$

The adjacency matrix A^r reflects the degree of semantic correlation between two regions in the feature map, based on which we select the top-k relevant regions for each region. Specifically, we obtain the routing index matrix $I^r \in N^{S^2 \times k}$ by performing the top-k operation row-by-row on A^r:

$$I^r = topkIndex\,(A^r) \tag{8}$$

The i-th row of I^r contains the index of the k regions that are most related to the i-th region. For each query token in the region i, we use the attention mechanism to attend to all key-value pairs that are located in the union of the top-k routed regions, denoted by $I^r_{(i,1)}, I^r_{(i,2)}, \ldots, I^r_{(i,3)}$:

$$K^g = gather\,(K, I^r) \tag{9}$$

$$V^g = gather\,(V, I^r) \tag{10}$$

where $K^g, V^g \in \mathbb{R}^{S^2 \times \frac{kHW}{S^2} \times C}$ are the key-value features we gathered, and finally we can perform attention calculation on these features, thus obtain the feature Out :

$$Output = Softmax\left(\frac{Q \times (K^g)^T}{\sqrt{C}}\right) \times V^g \tag{11}$$

IV. Experiments

A. Implementation Details

We adopt the Pytorch framework to implement our proposed method and train the model on an NVIDIA 3060 GPU. The input image size of our method is 224×224, and the backbone network is EfficientNet. We use the AdamW to optimize our network and choose the binary cross-entropy loss as our loss function. We set the initial learning rate to 0.0004. In the process of training, the learning rate will be reduced by 20% every two epochs. The minimum learning rate is set to 0.00004. The batch size is set to 12 and the model is trained for 50 epochs. The training parameters for our method are set at 17.6 M, and the training time for 50 epochs is approximately 43 hours. To evaluate our method, we utilize current mainstream face forgery detection datasets, such as FaceForensics++ (FF++) [8], FaceShifter [28], Deepfake Detection Challenge dataset (DFDC) [29], Celeb-DF [30] and WildDeepfake(Wild-DF) [31].

TABLE III
Cross-method evaluation results on FF++ (LQ) dataset.

Training Set	Methods	Testing Set (AUC)			
		DF	F2F	FS	NT
DF	EfficientNet [2]	98.63%	60.55%	59.65%	60.87%
	GramNet [16]	98.19%	57.05%	62.80%	57.04%
	GFFD [23]	97.87%	61.09%	59.57%	**62.41%**
	Ours	**98.74%**	**62.17%**	**64.10%**	61.84%
F2F	EfficientNet [2]	**70.60%**	95.10%	58.21%	**61.74%**
	GramNet [16]	61.26%	92.77%	54.19%	58.73%
	GFFD [23]	68.35%	93.32%	55.63%	61.31%
	Ours	67.30%	**95.41%**	**58.45%**	59.36%
FS	EfficientNet [2]	71.90%	**59.79%**	97.98%	51.04%
	GramNet [16]	73.43%	57.56%	96.15%	51.93%
	GFFD [23]	66.80%	54.97%	96.31%	51.14%
	Ours	**77.73%**	56.55%	**98.15%**	**53.20%**
NT	EfficientNet [2]	71.92%	68.19%	51.62%	86.89%
	GramNet [16]	68.67%	69.35%	48.55%	83.44%
	GFFD [23]	74.47%	67.19%	53.66%	83.96%
	Ours	**74.91%**	**69.71%**	**53.75%**	**87.23%**

B. Performance Evaluation

1) Evaluation on FaceForensics++ dataset: We conduct experiments on the HQ and LQ subsets of the FaceForensics++ dataset. The experimental results are listed in Table I. The performance of our proposed method is significantly better than other detection methods. In the HQ dataset, the videos have high definition, and retain rich information for detection due to the high image quality, making other methods effective. However, our proposed method still achieve the maximum Area Under Curve (AUC) of 98.78% and an Accuracy (ACC) of 94.14%. Compared to the HQ dataset, the heavily compressed LQ videos lose some low-level tampering artifacts, making detection more challenging. As seen in Table I, the performance of all methods drops significantly in this dataset, but our method still outperforms the others, with an ACC of 79.58% and an AUC of 89.93%.

2) Evaluation on FaceShifter dataset: FaceShifter [28] is a dataset of a single tampering method. There are some special artifacts in the image that can be captured by CNNs. Therefore, most detection methods have high classification performance. The prediction results are reported in Table I. Our proposed method still achieve the best performance. The AUC of HQ subset and LQ subset are respectively 99.83% and 98.48%.

TABLE IV
ABLATION RESEARCH OF SEVERAL MODULES ON FF++ (LQ).

Models	ACC	AUC
Spatial Stream	76.60%	89.09%
Noise Stream	77.87%	89.19%
Noise Stream + LEAM	76.87%	89.36%
Spatial Stream + LEAM	77.54%	89.66%
Spatial + Noise + DFFM	77.03%	89.83%
Spatial + Noise + DFFM + LEAM	**79.59%**	**89.93%**

This experimental data demonstrates the effectiveness of our proposed method.

TABLE V
COMPUTATIONAL COMPLEXITY COMPARISON RESULTS IN TERMS OF PARAMS AND FLOPS.

Methods	Params	FLOPs
EficientNet-B4 [2]	17.5 M	1.6 G
Xception [1]	22.1 M	4.6 G
GFFD [23]	53.2 M	13.8 G
GramNet [16]	11.7 M	2.3 G
M2TR [21]	38.0 M	4.6 G
Ours	17.6 M	2.0 G

C. Generalization Ability Evaluation

1) Generalize from one dataset to another: To verify the generalization performance of our proposed method, We train the models on the HQ subset of the FF++ dataset and evaluate the models on the test sets of Celeb-DF, DFDC and Wild-DF, and the results are shown in Table II. For the Celeb-DF, DFDC and Wild-DF datasets, it can be observed that the detection results of all methods have a significant decline. We think this is because the differences of forgery methods, and the more complex scenes. Nonetheless, our proposed method outperforms other methods, indicating that combining high-frequency noise with spatial features can effectively improve the generalization ability of the model.

2) Generalize from one method to another: Since the generation ability is crucial for face forgery detection tasks, we conduct a series of generalization experiments on the FF++ (LQ) dataset, which contains fake images generated by four different tampering techniques: Deepfakes (DF), Face2Face (F2F), FaceSwap (FS), and NeuralTextures (NT). We select one of the methods for training and test on other methods. The experimental results are shown in Table III. We can observe that the generalization performance of our proposed method is generally better than EfficientNet, GramNet and GFFD. This is because our method utilizes spatial information and high-frequency noise in the image, and enhances the fused features.

D. Ablation Study

In order to systematically illustrate the effectiveness of each module in our proposed method, we conduct a series of experiments on the FF++ (LQ) dataset, with the AUC metric used to evaluate performance. The experimental results are listed in Table IV. Spatial stream refers to using the RGB image as the input for the backbone network, while Noise stream refers to using high-frequency noise information for supplementation. Use of the Noise stream alone means the noise image was used as the input for the backbone network. The experiments reveal that spatial and noise information together yielded better results than using either alone, confirming the complementary nature of the two modalities. As shown in Table IV, as we add different modules to the network, the AUC metric keeps improving. Experimental results demonstrate the effectiveness of our proposed modules.

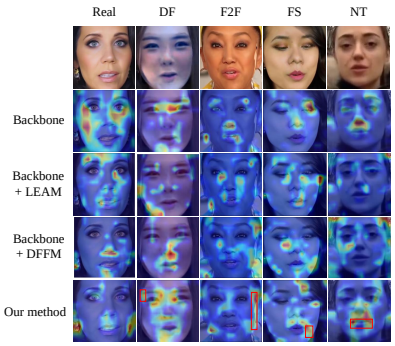

Fig. 4. The Grad-CAM visualization for forged faces on FF++ dataset, and the red boxes denote areas that previous methods failed to notice.

E. Evaluation of computational cost

We conduct a comparison of computational cost to gain a deeper understanding of our method's performance in detection efficiency. Specifically, we analyze the model's number of parameters (Params) and the number of floating-point operations (FLOPs), two metrics that directly reflect the neural network model's complexity in both space and time. As shown in the Table V, our comparison includes commonly used baseline networks in DeepFake detection, such as EfficientNet and Xception, as well as advanced detection methods like GramNet, M2TR, and GFFD. To ensure fairness in the comparison, all evaluations are conducted under the same experimental conditions, with the input image size set at 224×224×3. The data from the table reveal that, since the baseline networks serve only as classification models without any reconstruction framework or additional modules designed specifically for DeepFake detection, they have lower parameter counts and floating-point operation numbers, which also account for their poor performance in Deepfake detection. Compared to EfficientNet, our model shows an increase of 0.1M in Params and 0.4G in FLOPs, primarily due to the introduction of our new Dual Feature Fusion Modul and Local Enhancement Attention Modul. However, in comparison to GramNet, M2TR, and GFFD, our method requires fewer parameters and FLOPs, demonstrating an advantage in computational efficiency.

F. Visualization

To observe the regions where our proposed method pays attention when facing different tampering methods, we employ

Gradient-weighted Class Activation Mapping (Grad-CAM) [32] to generate heat maps presented in Fig. 4. In these heat maps, warm color denotes the regions that are sensitive to the prediction. Trained on FF++(HQ), both Backbone and our proposed method can effectively concentrate on the tampered regions of faces created by the four tampering techniques. And we also visualize the DFFM and LEAM proposed in this article, as shown in Fig. 4, our method demonstrates a more comprehensive focus area when we adopt two modules.

V. Conclusions

In this paper, we identify forgery traces that are often challenging to detect in the noise space. These local noise inconsistencies are found to be complementary to the tampering artifacts in spatial domain. Based on this, we propose a novel face forgery detection method and design two functional modules: Dual Feature Fusion Module and Local Enhanced Attention Module. These modules help us extract more informative features and effectively leverage the complementarity between different modal features. To evaluate the performance of our proposed method, we conduct comprehensive experiments on various mainstream datasets. Compared to the competing methods, our approach demonstrate outstanding forgery detection and generalization abilities.

References

[1] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 1251–1258.

[2] M. Tan and Q. Le, "Efficientnet: Rethinking model scaling for convolutional neural networks," in *International Conference on Machine Learning*, 2019, pp. 6105–6114.

[3] H. Chen, Y. Lin, B. Li, and S. Tan, "Learning features of intra-consistency and inter-diversity: Keys toward generalizable deepfake detection," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 33, no. 3, pp. 1468–1480, 2022.

[4] Y. Yu, R. Ni, Y. Zhao, S. Yang, F. Xia, N. Jiang, and G. Zhao, "Msvt: Multiple spatiotemporal views transformer for deepfake video detection," *IEEE Transactions on Circuits and Systems for Video Technology*, pp. 4462–4471, 2023.

[5] X. Li, R. Ni, P. Yang, Z. Fu, and Y. Zhao, "Artifacts-disentangled adversarial learning for deepfake detection," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 33, no. 4, pp. 1658–1670, 2022.

[6] J. Yang, S. Xiao, A. Li, W. Lu, X. Gao, and Y. Li, "Msta-net: Forgery detection by generating manipulation trace based on multi-scale self-texture attention," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 32, no. 7, pp. 4854–4866, 2021.

[7] Z. Shang, H. Xie, L. Yu, Z. Zha, and Y. Zhang, "Constructing spatio-temporal graphs for face forgery detection," *ACM Transactions on the Web*, vol. 17, no. 3, pp. 1–25, 2023.

[8] A. Rossler, D. Cozzolino, L. Verdoliva, C. Riess, J. Thies, and M. Nießner, "Faceforensics++: Learning to detect manipulated facial images," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 1–11.

[9] Y. Qian, G. Yin, L. Sheng, Z. Chen, and J. Shao, "Thinking in frequency: Face forgery detection by mining frequency-aware clues," in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XII*. Springer, 2020, pp. 86–103.

[10] J. Li, H. Xie, L. Yu, X. Gao, and Y. Zhang, "Discriminative feature mining based on frequency information and metric learning for face forgery detection," *IEEE Transactions on Knowledge and Data Engineering*, vol. 35, no. 12, pp. 12 167–12 180, 2021.

[11] P. Zhou, X. Han, V. I. Morariu, and L. S. Davis, "Learning rich features for image manipulation detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 1053–1061.

[12] X. Wu, Z. Xie, Y. Gao, and Y. Xiao, "Sstnet: Detecting manipulated faces through spatial, steganalysis and temporal features," in *ICASSP 2020-2020 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE, 2020, pp. 2952–2956.

[13] A. Haliassos, K. Vougioukas, S. Petridis, and M. Pantic, "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*, 2021, pp. 5039–5049.

[14] T. Jung, S. Kim, and K. Kim, "Deepvision: Deepfakes detection using human eye blinking pattern," *IEEE Access*, vol. 8, pp. 83 144–83 154, 2020.

[15] D. Afchar, V. Nozick, J. Yamagishi, and I. Echizen, "Mesonet: a compact facial video forgery detection network," in *2018 IEEE International Workshop on Information Forensics and Security*. IEEE, 2018, pp. 1–7.

[16] Z. Liu, X. Qi, and P. H. Torr, "Global texture enhancement for fake face detection in the wild," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 8060–8069.

[17] M. Yu, J. Zhang, S. Li, and J. Lei, "Msfrnet: Two-stream deep forgery detector via multi-scale feature extraction," *IET Image Processing*, vol. 17, no. 2, pp. 581–596, 2023.

[18] L. Zhao, M. Zhang, H. Ding, and X. Cui, "Mff-net: Deepfake detection network based on multi-feature fusion," *Entropy*, vol. 23, no. 12, p. 1692, 2021.

[19] S. Liu, Q. Jiang, X. Jin, Z. He, W. Zhou, S. Yao, and Q. Wang, "Multiple feature mining based on local correlation and frequency information for face forgery detection," in *2022 IEEE 34th International Conference on Tools with Artificial Intelligence (ICTAI)*. IEEE, 2022, pp. 1347–1354.

[20] C. Lin, F. Yi, H. Wang, Q. Li, D. Jingyi, and C. Shen, "Exploiting facial relationships and feature aggregation for multi-face forgery detection," *arXiv preprint arXiv:2310.04845*, 2023.

[21] J. Wang, Z. Wu, W. Ouyang, X. Han, J. Chen, Y.-G. Jiang, and S.-N. Li, "M2tr: Multi-modal multi-scale transformers for deepfake detection," in *Proceedings of the 2022 International Conference on Multimedia Retrieval*, 2022, pp. 615–623.

[22] J. Wu, B. Zhang, Z. Li, G. Pang, Z. Teng, and J. Fan, "Interactive two-stream network across modalities for deepfake detection," *IEEE Transactions on Circuits and Systems for Video Technology*, pp. 6418–6430, 2023.

[23] Y. Luo, Y. Zhang, J. Yan, and W. Liu, "Generalizing face forgery detection with high-frequency features," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 16 317–16 326.

[24] Z. Guo, G. Yang, D. Zhang, and M. Xia, "Rethinking gradient operator for exposing ai-enabled face forgeries," *Expert Systems with Applications*, vol. 215, p. 119361, 2023.

[25] C. Wang and W. Deng, "Representative forgery mining for fake face detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 14 923–14 932.

[26] H. Zhang, I. Goodfellow, D. Metaxas, and A. Odena, "Self-attention generative adversarial networks," in *International Conference on Machine Learning*. PMLR, 2019, pp. 7354–7363.

[27] L. Zhu, X. Wang, Z. Ke, W. Zhang, and R. W. Lau, "Biformer: Vision transformer with bi-level routing attention," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 10 323–10 333.

[28] L. Li, J. Bao, H. Yang, D. Chen, and F. Wen, "Faceshifter: Towards high fidelity and occlusion aware face swapping," *arXiv preprint arXiv:1912.13457*, 2019.

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

[30] Y. Li, X. Yang, P. Sun, H. Qi, and S. Lyu, "Celeb-df: A large-scale challenging dataset for deepfake forensics," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 3207–3216.

[31] B. Zi, M. Chang, J. Chen, X. Ma, and Y.-G. Jiang, "Wilddeepfake: A challenging real-world dataset for deepfake detection," in *Proceedings of the 28th ACM International Conference on Multimedia*, 2020, pp. 2382–2390.

[32] 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 *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 618–626.