Learning Local Reconstruction Errors for Face Forgery Detection

Haoyu Wu, Lingyun Leng, Peipeng Yu
College of Cyberspace Security, Jinan University, Guangzhou, China

***Abstract*—Although several deepfake detection technologies have achieved great detection accuracy inside the data domain in recent years, there are still limitations in cross-domain generalization. This is due to the model's ease of fitting the data sample distribution in the training data domain and its tendency to detect a specific forgery trace in order to reach a judgment rather than catching generalized forgery traces. In this paper, we propose to learn Local Reconstruction Errors for face forgery detection. The local anomaly traces of the fake face are often mapped using the original real face as a reference; however, the original real face of the fake face cannot be acquired in the real scenario. Therefore, this solution designs a local reconstruction autoencoder trained with real samples. By masking key areas of the face, the original real face can be reconstructed. Because the autoencoder only learns how to restore the essential parts of the real face using local patches of real samples, it cannot recover the forging traces or target face information in the fake face. Therefore, the reconstructed image forms a reconstructed difference with the original image. This solution aids the model in detecting local differences in fake faces by producing feature-level local difference attention mappings in the network's middle layer. A series of experiments demonstrate that this solution has good detection and generalization performance.**

Keywords*—*Face forgery; deepfake detection; local anomalies; generalized detection

I. Introduction

With the rapid development of deep learning technology, deepfake technology has found widespread applications. By manipulating or replacing images or videos of faces, deepfake technology can alter visual content in subtle ways, posing significant threats to privacy, public opinion, and information security[1], [2], [3], [4]. Consequently, effectively detecting forged faces has become a crucial research topic in the field of computer vision.

Currently, there have been significant advancements in deepfake detection, with many methods performing well on forged data similar to their training datasets [5], [6], [7]. However, these methods often lack generalization when faced with unknown types of forgeries. Enhancing the generalization of detection models across various forgery methods is an urgent challenge. Our primary motivation is that there are notable differences between forged faces and their authentic counterparts (in terms of identity, artifacts, etc.). By leveraging these differences, we can accurately identify key areas of forgery rather than merely learning a single forgery pattern. Traditional methods either require reference images of the original faces or use self-attention mechanisms to predict key areas, both of which have significant limitations.

With the above considerations in mind, in this paper, we propose a local reconstruction-based deepfake detection method. By designing a local reconstruction autoencoder trained on real samples, we can mask and reconstruct critical regions of the face. The model generates a reconstruction difference map between the forged and real faces. Since the autoencoder cannot reconstruct the forgery traces in the forged face, this reconstruction difference map provides new discriminative information for forgery detection. Furthermore, we introduce a feature-level local difference attention map within the model to enhance the focus on forged regions. A series of experimental results demonstrate that this approach exhibits excellent detection performance and generalization capability across multiple datasets.In brief, our contributions are summarized as follows:

- We propose a novel detection framework based on local reconstruction for restoring genuine faces, which can eliminate artifacts in forged faces and guide the model to learn key regions.
- We introduce a local reconstruction autoencoder framework based on a key region masking algorithm, capable of restoring the original genuine face from local genuine patches.
- We present a method that uses local feature attention maps based on reconstructed image comparison to guide the detection model to focus on key regions and learn highly generalizable features.
- Our approach effectively enhances the generalization ability of the detection model on unknown datasets and against unknown forgery methods.

II. Related Work

A. Face Forgery Algorithms

Recent face forgery methods benefit from advances in deep learning. It can be classified into three categories based on the target of manipulation: face swapping, face editing, and face generation. In the early stages, researchers [3] viewed face swapping as a style transfer problem. Guided by facial landmark points, convolutional neural networks (CNNs) could transform one facial image into another, adopting the style of a face with a specific identity. However, with the rapid advancement of deep learning, several novel face swapping algorithms have emerged, significantly reducing the difficulty of face swapping [8], [9]. The progress of Generative Adversarial Networks (GANs) has further enhanced the realism of forged faces [10], [11], [12].

B. Face Forgery Detection Algorithms

Deepfake technology often produces noticeable artifacts when synthesizing or distorting facial features, such as unreasonable distortions of facial elements, edge artifacts, and missing details. Matern *et al.* [13] observed that certain Deepfake and Face2Face forgeries resulted in visual anomalies like differences in eye color, distorted facial contours, and missing tooth details. They aimed to detect these inconsistencies, but such artifacts only appear in lower-quality deepfake products, lacking universality. Nirkin *et al.* [14] proposed a method that segments detection images into internal (eyes, nose, mouth) and external (ears, hair) facial regions to train separate vectors for feature extraction. However, this approach does not adapt well to forgeries affecting external facial areas, resulting in limited generalization. Liu *et al.* [15] identified fundamental statistical differences in texture data between forged and real faces, leading to the development of a novel architecture for global texture representation to enhance the robustness of forgery detection. Chen *et al.* [16] used facial masking to detect whether images had undergone interference, reconstructing affected images to check for artifacts in the cleaned results. Dong *et al.*[17] approached this through image matching, proposing that forged images contain artifacts unrelated to the features of the original and target images. They designed a training set of matching images (forged, original, and target) to implicitly guide model learning, achieving good performance against compression. Wang *et al.* [18] developed a deepfake detection model focused on identifying potential noise traces, extracting features from both facial and background segments. They employed a novel multi-head contrastive interaction method to assess the similarity between facial and background noise features for image authenticity detection. Huang *et al.* [19] highlighted the differences between explicit and implicit identities in swapped images, introducing explicit identity contrast loss and implicit identity exploration loss to increase the distance between the explicit and implicit identities of fake faces, using this information for authenticity determination. However, Dong *et al.* [20] argued that focusing on identity information hinders the generalization of classification models, leading to a breakthrough with a method that prioritizes local features while ignoring overall identity information.

In summary, deepfake detection algorithms are designed to guide models in capturing specific artifact features, thereby identifying manipulated images by responding to these artifacts. Nevertheless, a common limitation of these methods is their often inadequate generalization capability when encountering previously unseen types of forgeries.

III. Methodology

A. Overview

The distribution of real human faces is consistent and uniform [21], while it is difficult for forged faces to completely eliminate all traces of artifacts, leading to a lack of continuity in the distribution of fake faces. Therefore this paper explores whether it is possible to design a method that constrains the model to learn key difference areas. A promising approach is to use the local differences between forged faces and their original faces, employing the original face to create an attention mask for the forged face. However, in most cases, the detection side cannot obtain the original face corresponding to the forged face. Thus, this paper attempts to reconstruct the distribution of the real face from the forged face to assist in detecting authenticity.

Based on the above concept, this paper proposes a deepfake detection scheme based on local reconstruction. As shown in Fig. 1, the scheme consists of two stages: (1) Training stage based on masked reconstruction of real samples. The content of a forged face typically originates from a source image and a target image, corresponding to its internal and external face, whereas a real image has a unified internal and external face. Therefore, this scheme pre-trains an encoder and decoder based on a Vision Transformer (ViT) using real face images. By randomly masking most facial features during face reconstruction, the encoder and decoder learn the distribution of real faces, acquiring the ability to recover the original real face information from partially masked real faces. (2) Training stage based on local difference attention map constraints. After training the encoder and decoder, their weights are fixed, and the to-be-detected image is masked and reconstructed to obtain a reconstructed image. Both the reconstructed image and the to-be-detected image are input into a fixed-weight feature extraction network to calculate multiple feature-level local difference attention maps. These attention maps are used to guide the model to learn key regional features for generalized detection.

B. Training Stage Based on Masked Reconstruction of Real Samples

The purpose of the training stage based on masked reconstruction of real samples is to train the encoder and decoder to learn the distribution of real face data, enabling the reconstruction of a complete real face image from local real face regions. Inspired by the MAE method, this scheme trains a ViT-based encoder and decoder structure using real faces with masked key facial regions. First, a face image of size $C \times H \times H$ is divided into N patches, each of size $C \times P \times P$, where $N = \frac{H^2}{P^2}$, and C, H, and P represent the number of image channels, the image's side length, and the patch's side length, respectively. Then, the proposed key region masking algorithm randomly masks patches in key facial regions, and the model learns to reconstruct the masked patches based on the remaining parts of the face image. The trained encoder and decoder learn the ability to recover the original real face information from local real face regions, allowing for effective reconstruction of the masked parts of a real face. When the encoder and decoder, trained solely on real face data, are used to perform masked reconstruction on a forged face, they can reconstruct the original real face from the genuine local regions of the forged face, but they will not restore the local forged artifacts in the fake face.

1) Key Region Masking Algorithm: Typically, the facial features, such as the eyes, nose, and mouth, are the key areas in forged faces. To ensure that the encoder and decoder learn to reconstruct the original face from real facial regions, and avoid reconstructing forged areas during fake face testing, this scheme defines a key region that includes facial features. As shown in Fig. 2, the proposed method first uses a landmark algorithm to extract the coordinates of 68 key facial points for all faces in the dataset. From these, it selects the coordinates

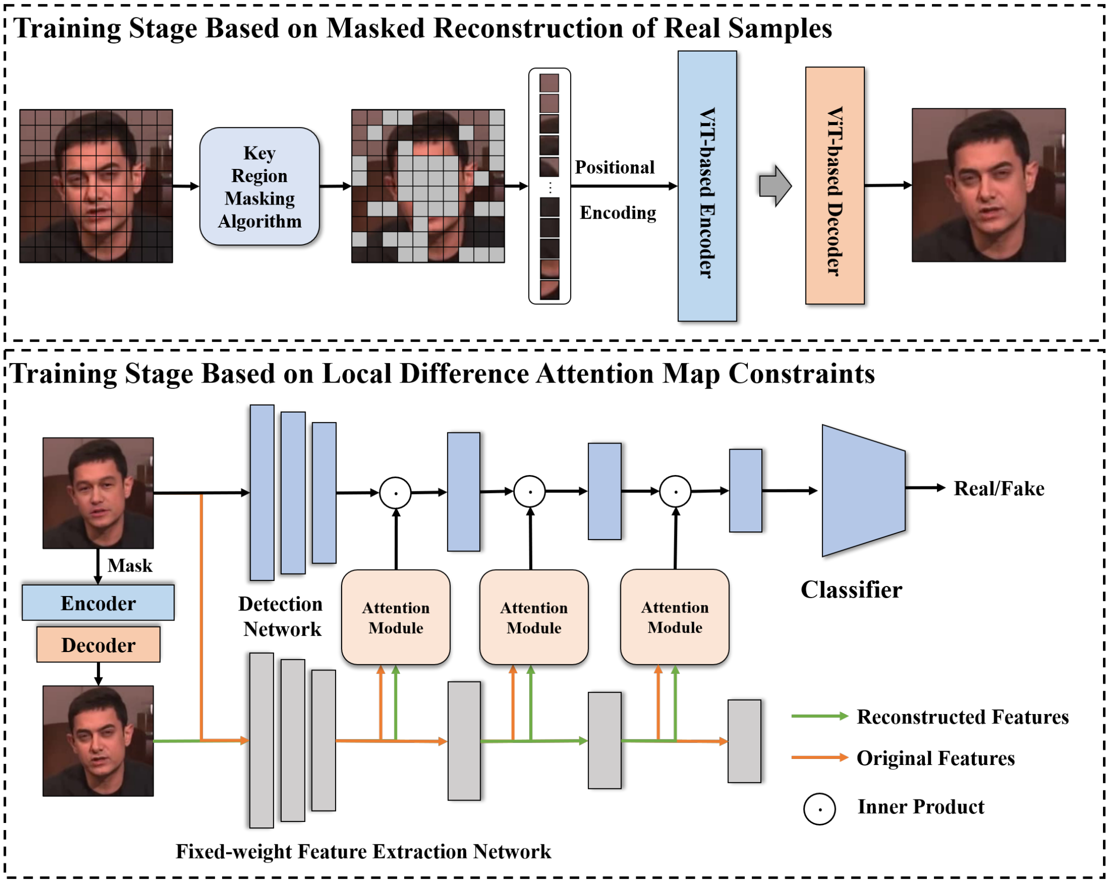

Fig. 1. Framework of local reconstruction-based deepfake detection algorithm.

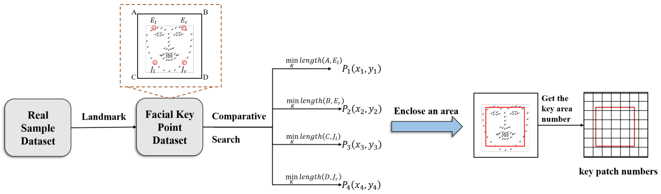

Fig. 2. Key area masking algorithm flowchart.

of the left eyebrow, right eyebrow, left jaw, and right jaw that are closest to the image vertices, denoted as P_1 , P_2, P_3, and P_4. The area enclosed by these points can cover the facial features of most faces in the dataset. Next, the four coordinates are expanded outward to form a rectangular region, which is defined as the key facial region. The face image is then divided into patches, where each small patch is assigned a number i, where $i \in \{1, 2, \ldots, N\}$. Based on the pixel coordinates of the key region, a set of key patch numbers, denoted as $T = \{45, 46 \ldots\}$, can be calculated, representing the patches included in the key region. Finally, random masking is applied to the image at a proportion of p, ensuring that most patches in the key patch set are masked. The sequence of unmasked patches is then fed into the encoder to reconstruct the original face image.

2) *Face Reconstruction via Encoder and Decoder:* The ViT-based encoder first receives the input sequence of unmasked patches and assigns a positional index to each unmasked patch. These patches are then passed through a series of Transformer blocks to learn the deep features of the real patch regions, which are used for subsequent face reconstruction. After undergoing a series of encoding processes, the encoder outputs the features of the unmasked patches. At this stage, the decoder takes these unmasked patch features as input and adds mask tokens to the masked areas to form a complete image, then applies positional encoding to all patch features. The mask tokens are shared, learnable vectors used to represent the masked patches that need to be reconstructed. Through a series of reconstruction processes within the decoder, the final linear layer of the decoder outputs a linear projection of the reconstructed image. After adjusting the dimensions and size, the reconstructed image is obtained. The reconstruction loss is then calculated only for the masked patches using mean squared error (MSE), with the loss expression as follows:

$$L_{rec} = \frac{1}{N} \sum_{i=1}^{n} (y_i - x_i)^2 \tag{1}$$

where x represents the reconstructed masked patches, and y represents the actual masked patches. By leveraging the information of masked patch indexes and calculating the reconstruction loss only for the masked patches, the computation is reduced, significantly improving the training efficiency of the

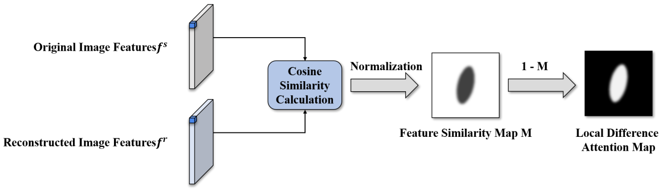
Original Image Featuresf^s
Reconstructed Image Featuresf^r
Cosine Similarity Calculation
Normalization
Feature Similarity Map M
1 - M
Local Difference Attention Map

Fig. 3. Feature-Level local difference attention map calculation flowchart.

encoder and decoder.

C. Training Stage Based on Local Difference Attention Map Constraints

The training stage based on local difference attention map constraints aims to guide the model to learn the local differences in forged face images by calculating feature-level local difference attention maps between the reconstructed and original images. After the ViT-based encoder and decoder are trained on real face data, they have only acquired the knowledge of reconstructing local real facial regions. Therefore, this scheme fixes the parameters of the encoder and decoder and applies them during the training phase of the detection task. First, after dividing the face image into patches and masking the key regions, the unmasked patch sequence is input into the encoder-decoder framework to obtain the reconstructed face image. Next, a pre-trained feature extraction network with fixed parameters is used to extract a series of feature maps from the middle layers of the network for both the reconstructed face and the original face. Using similarity calculations, feature-level local difference attention maps are obtained. To ensure that the size of the feature-level local difference attention maps matches the feature maps in the middle layers of the detection network, both the feature extraction network and the detection network adopt the Xception architecture. The method for calculating the local difference attention map is shown in Fig. 3. This scheme uses cosine similarity to compute the similarity at each location between the original image features and the reconstructed image features. After normalizing the results, by subtracting the feature similarity map M from 1 to emphasize the difference regions, the final local difference attention map is obtained. The calculation expression is as follows:

$$AttentionMap_i = 1 - \frac{1 + CosineSimilarity(f_i^s, f_i^r)}{2} \tag{2}$$

where f^s and f^r represent the features of the original image and the reconstructed image, respectively, and $CosineSimilarity$ represents the cosine similarity calculation, which ranges from $[-1, 1]$. The higher the value, the more similar the two features are.

In the final classification prediction, this scheme uses the same network model as the feature extraction network described above as the basic framework, allowing the local difference attention map to guide the model in learning the key regional difference features. As shown in Fig. 1, the proposed method performs an inner product between the multiple feature maps of the original face image from the middle layers of the network and the local difference attention map, thereby constraining the model's learning process. Finally, the features extracted by the model are input into the classifier for real/fake classification, and binary cross-entropy (BCE) loss is used to constrain the training. The loss function is as follows:

$$L_{cls} = -\frac{1}{N}\sum_{k=1}^{n} y_k \log(x_k) + (1 - y_k)\log(1 - x_k) \tag{3}$$

where x represents the real/fake prediction, and y represents the real/fake label.

TABLE I. RESULTS OF IN-DATASET EVALUATIONS

Methods	**FF++(C23)**		**FF++(C40)**	
	Acc	**AUC**	**Acc**	**AUC**
Face-X-ray[22]	—	87.4	—	61.6
MesoNet[5]	83.1	84.3	70.47	72.62
Multi-task[6]	85.65	85.43	81.3	75.59
XceptionELA[23]	93.86	94.8	79.63	82.9
SPSL[24]	91.5	95.32	81.57	82.82
CFFs[25]	—	97.21	—	86.56
M2TR[26]	91.86	96.75	83.89	87.15
Two-branch[27]	96.43	98.7	86.34	86.59
HFI-Net[28]	91.87	97.07	58.69	88.4
RFM[29]	**95.69**	**98.79**	**87.06**	89.83
Ours	91.78	97.4	80.75	**90.12**

IV. EXPERIMENTS

A. Datasets

The experiments in this study utilize the following three datasets for testing and evaluation: FaceForensics++[30], Celeb-DF-v2[31], and the DFD dataset[32]. FaceForensics++ is a large public dataset for facial forgery detection, containing 1,000 real videos and 4,000 forged videos generated using four manipulation methods: Deepfakes, Face2Face, FaceSwap, and NeuralTextures. Additionally, FaceForensics++ includes three compression levels: the original version (C0), a high-quality version (C23), and a low-quality version (C40). Celeb-DF-v2 is a challenging dataset composed of 569 real videos and 5,639 forged videos extracted from YouTube. The DFD dataset is another large-scale dataset containing 363 real videos and 3,068 forged videos across various scenarios.

B. Experimental Setup

The experiments in this study are implemented using the PyTorch framework, with programming conducted in Python. The datasets are divided for training, validation, and testing of the detection model. OpenCV is used to extract a series of continuous, non-repeating video frames from videos at fixed intervals. The RetinaFace face recognition algorithm is employed to locate the face regions in the video frames, align these regions, and crop them appropriately. All face images are resized to a uniform dimension of 224×224 pixels. All experiments utilize the Adam optimizer for training, with a learning rate set to 0.0001 and a batch size of 32. The ViT-based encoder and decoder are trained for 300 epochs, with a masking ratio of 75%. The Xception-based feature extraction network and classifier are trained for 30 epochs, with 200 iterations per epoch. The training is conducted on

TABLE II. RESULTS OF CROSS-MANIPULATION EVALUATIONS ON FF++C23(AUC)

Methods	**Train**	**DF**	**F2F**	**FS**	**NT**	**Avg**
En-b4[33]	**DF**	99.65	**73.6**	40.73	**73.94**	71.98
SimMIM[34]		99.64	62.43	66.74	62.74	**72.89**
FDFL[34]		98.91	58.9	**66.87**	63.61	72.07
Ours		**99.85**	70	41.38	71.42	70.66
En-b4[33]	**F2F**	**87.15**	99.26	51.6	66.85	**76.22**
SimMIM[34]		84.27	**99.28**	53.49	53.87	72.73
FDFL[34]		67.55	93.06	55.35	66.66	70.66
Ours		78.99	99.01	**55.53**	**70.45**	75.92
En-b4[33]	**FS**	61.44	68.96	**99.57**	49.83	69.95
SimMIM[34]		**88.12**	58.88	99.19	**52.55**	**74.67**
FDFL[34]		75.9	54.64	98.37	49.72	69.66
Ours		63.4	**70.79**	99.53	51.48	71.3
En-b4[33]	**NT**	83.98	69.08	46.32	**97.59**	74.24
SimMIM[34]		**85.26**	64.38	46.62	69.95	73.38
FDFL[34]		79.09	74.21	**53.99**	88.54	73.96
Ours		84.14	**76.4**	50.88	96.51	**76.98**

an NVIDIA GTX GeForce 3090 Ti platform with 24 GB of VRAM. Additionally, binary classification accuracy (Acc) and the area under the ROC curve (AUC) are used as performance evaluation metrics for the model.

C. In-Dataset Evaluation

This section tests the in-dataset detection performance of the proposed method on the FaceForensics++ (FF++) datasets, and compares it with other state-of-the-art methods. The proposed method is independently trained on and validated using the test sets of FF++(C23) and FF++(C40) datasets. The results are shown in Table I. It can be observed that, the proposed method achieves high test accuracy on the FF++(C23) datasets. The lower AUC performance on FF++(C40) may be attributed to the inconsistent quality of the original compressed images. During the training phase with real samples, the encoder and decoder did not incorporate lower-quality images for training, resulting in deviations and quality issues when reconstructing low-quality images.

D. Cross-Dataset Evaluation

1) Cross-Manipulation Method Evaluation: Cross-manipulation method evaluation is a significant approach to assess the generalization capability of detection methods, with important practical implications. This section conducts cross-testing of the proposed method across four different manipulation methods on the FF++(C23) dataset, with the results shown in Table II. It can be observed that the average performance across the four cross-tests exceeds 70%. In comparison with other advanced methods, the proposed approach, trained on the NT dataset, achieves a higher average AUC performance of 76.98% across the four manipulation methods, representing an improvement of 2% in detection performance. Additionally, the average test results from training on F2F reach 75.92%, with a gap of less than 1% compared to the higher performance of En-b4. The experimental data in the table demonstrate that the proposed method exhibits effective generalization across single-source manipulation methods, confirming the feasibility of this approach.

The aforementioned experiments demonstrate the generalization evaluation from a single manipulation domain to other domains. Additionally, there exists a method for evaluating generalization in multi-source forgery detection. This experiment utilizes three training sets from FF++ (excluding DF) for joint training and tests on DF, defined in the table as GID-DF. Similarly, the experiment trains on three other manipulation sets (excluding F2F) and tests on F2F. All test results are presented in Table III. For the DF tests, existing methods have reached a high performance level, with the proposed method closely following, showing an AUC performance difference of less than 7% from the state-of-the-art methods. Although there is a gap in AUC performance for GID-DF(C23) compared to the leading methods, the accuracy performance and results on the C40 version dataset remain outstanding, surpassing other existing advanced methods. Regarding the F2F tests, mainstream methods show subpar performance, while the proposed method achieves the best results, exceeding current advanced methods by 1% in AUC performance and 2% in accuracy performance, demonstrating improvements across different compression levels of F2F images. These experimental data strongly affirm the superiority of the proposed method in multi-source forgery detection.

2) Cross-Dataset Evaluation: This section evaluates the performance of the proposed method across different datasets. The method was trained on the FF++(C23) dataset and tested on the FF++(C23) library, Celeb-DF-v2, and DFD. The experimental results, as shown in Table IV, indicate that the method achieved the best AUC performance on the DFD dataset and demonstrated highly competitive performance on Celeb-DF-v2, surpassing most advanced methods with a gap of less than 5% compared to the state-of-the-art methods. This suggests that the proposed method exhibits good generalization capabilities across unknown datasets.

E. Ablation Study

As the proposed method is an integrated detection framework, it is not possible to conduct an ablation study on individual components or stages. Here, we present the comparative experimental results between the proposed method and the Xception-based classification baseline model. The baseline model was trained on FF++(C23) and tested for AUC performance on Celeb-DF-v2 and DFD. The testing results, as shown in Table V, indicate that the proposed method outperforms the baseline model on both Celeb-DF-v2 and DFD, demonstrating its effectiveness.

F. Visualization of Results

This section further demonstrates the reconstructed images generated by the autoencoder framework trained on real samples. As shown in Fig. 4, the similarity between the real image and the reconstructed image is very high for real face images. However, for forged images, which contain artifacts not present in real samples, reconstructing the forged images with masked key facial regions results in reconstructed images that do not retain the original forged information, leading to significant local differences compared to the original forged images. Additionally, since the encoder and decoder are capable of reconstructing real samples, the reconstructed facial features of the forged images tend to resemble those of the original

TABLE III. Results of Multi-source Manipulation Evaluations on FF++

Methods	GID-DF(C23)		GID-DF(C40)		GID-F2F(C23)		GID-F2F(C40)	
	Acc	AUC	Acc	AUC	Acc	AUC	Acc	AUC
EfficientNet[33]	82.4	91.11	67.6	75.3	63.32	80.1	61.41	67.4
Focalloss[35]	81.33	90.31	67.47	74.95	60.8	79.8	64	67.21
ForensicTransfer[36]	72.01	—	68.2	—	64.5	—	55	—
Multi-task[6]	70.3	—	66.76	—	58.74	—	56.5	—
MLDG[37]	84.21	91.82	67.15	73.12	63.46	77.1	58.12	61.7
LTW[38]	85.6	92.7	69.15	75.6	65.6	80.2	65.7	72.4
DCL[39]	**87.7**	**94.9**	**75.9**	**83.82**	68.4	82.93	67.85	75.07
Ours	79.28	87.9	70.22	78.67	**73.62**	**84.29**	**69.25**	**76.5**

TABLE IV. Results of Cross-Dataset Evaluations on FF++C23(AUC)

Methods	FF++(C23)	Celeb-DF-v2	DFD
TI2Net[40]	**99.95**	68.22	72.03
FRLM[41]	99.5	70.58	68.17
F3Net[42]	98.1	71.21	86.1
Face-X-ray[22]	87.4	74.2	85.6
MLDG[37]	98.99	74.56	88.14
GFF[43]	98.36	75.31	85.51
SFDG[44]	99.53	75.83	88
SOLA[45]	99.25	76.02	—
MultiAtt[46]	99.27	76.65	87.58
BIG-Arts[47]	99.39	77.04	89.92
LTW[38]	99.17	77.14	88.56
FAAFF[48]	99.27	77.59	—
Local-Relation[49]	99.46	78.26	89.24
DCL[39]	99.3	**82.3**	91.66
Ours	97.24	77.47	**95.23**

TABLE V. Results of Ablation Study

Methods	FF++(C23)	Celeb-DF-v2	DFD
Baseline	**99.09**	72.15	87.86
Ours	97.24	**77.47**	**95.23**

real faces, which is influenced by the training effect of the encoder and decoder on large amounts of real face data.

V. Conclusion

This paper proposes a deepfake detection algorithm based on local reconstruction, comprising two stages: the training stage based on masked reconstruction of real samples and the training stage based on local difference attention map constraints. There are local differences between forged faces and the original real faces, and attention maps generated from these local differences can guide the model to learn key forgery regions, shifting the model's focus from global to local features to improve detection performance. Previous methods either require the original real image as a reference or use self-attention mechanisms to predict key regions, both of which have significant limitations. In contrast, this method enhances practical applicability by using a local reconstruction approach to recover the original real face from local regions of real faces, aligning better with real-world scenarios. By calculating feature-level local difference attention maps between the reconstructed and original images, the model is effectively constrained to learn the features of key forgery regions, further enhancing its ability to extract difference features. Extensive experiments demonstrate the effectiveness and reliability of this method in improving generalization performance. However, our local reconstruction method does not fully exploit the information available in forged faces and struggles with reconstructing low-quality faces. In future research, we aim to develop new algorithms that incorporate the identity information in forged faces to better recover the original real faces. In the future, leveraging local masking and reconstruction to restore real faces holds significant potential and valuable research implications for both generalized detection and proactive forensic analysis.

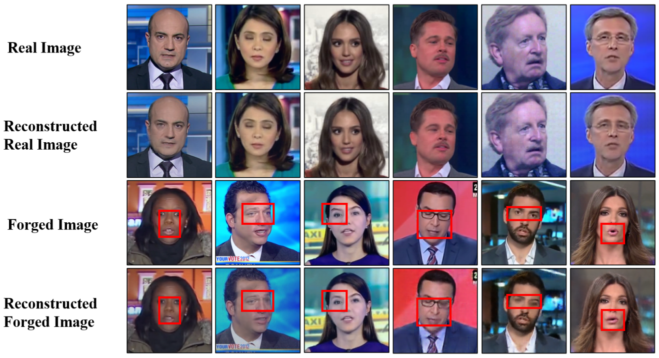

Fig. 4. Reconstruction results of real and forged images.

References

[1] T. Karras, "Progressive growing of gans for improved quality, stability, and variation," *arXiv preprint arXiv:1710.10196*, 2017.

[2] T. Karras, S. Laine, and T. Aila, "A style-based generator architecture for generative adversarial networks," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 4401–4410.

[3] I. Korshunova, W. Shi, J. Dambre, and L. Theis, "Fast face-swap using convolutional neural networks," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 3677–3685.

[4] J. Thies, M. Zollhofer, M. Stamminger, C. Theobalt, and M. Nießner, "Face2face: Real-time face capture and reenactment of rgb videos," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 2387–2395.

[5] 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 (WIFS). IEEE, 2018, pp. 1–7.

[6] H. H. Nguyen, F. Fang, J. Yamagishi, and I. Echizen, "Multi-task learning for detecting and segmenting manipulated facial images and videos," in *2019 IEEE 10th international conference on biometrics theory, applications and systems (BTAS)*. IEEE, 2019, pp. 1–8.

[7] H. H. Nguyen, J. Yamagishi, and I. Echizen, "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)*. IEEE, 2019, pp. 2307–2311.

[8] R. Natsume, T. Yatagawa, and S. Morishima, "Fsnet: An identity-aware generative model for image-based face swapping," in *Computer Vision–ACCV 2018: 14th Asian Conference on Computer Vision, Perth, Australia, December 2–6, 2018, Revised Selected Papers, Part VI 14*. Springer, 2019, pp. 117–132.

[9] Y. Nirkin, Y. Keller, and T. Hassner, "Fsgan: Subject agnostic face swapping and reenactment," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, pp. 7184–7193.

[10] Z. He, W. Zuo, M. Kan, S. Shan, and X. Chen, "Attgan: Facial attribute editing by only changing what you want," *IEEE transactions on image processing*, vol. 28, no. 11, pp. 5464–5478, 2019.

[11] H. Tang, D. Xu, N. Sebe, and Y. Yan, "Attention-guided generative adversarial networks for unsupervised image-to-image translation," in *2019 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2019, pp. 1–8.

[12] X. Li, S. Zhang, J. Hu, L. Cao, X. Hong, X. Mao, F. Huang, Y. Wu, and R. Ji, "Image-to-image translation via hierarchical style disentanglement," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 8639–8648.

[13] F. Matern, C. Riess, and M. Stamminger, "Exploiting visual artifacts to expose deepfakes and face manipulations," in *2019 IEEE Winter Applications of Computer Vision Workshops (WACVW)*. IEEE, 2019, pp. 83–92.

[14] Y. Nirkin, L. Wolf, Y. Keller, and T. Hassner, "Deepfake detection based on discrepancies between faces and their context," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 10, pp. 6111–6121, 2021.

[15] 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.

[16] Z. Chen, L. Xie, S. Pang, Y. He, and B. Zhang, "Magdr: Mask-guided detection and reconstruction for defending deepfakes," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 9014–9023.

[17] S. Dong, J. Wang, J. Liang, H. Fan, and R. Ji, "Explaining deepfake detection by analysing image matching," in *European conference on computer vision*. Springer, 2022, pp. 18–35.

[18] T. Wang and K. P. Chow, "Noise based deepfake detection via multi-head relative-interaction," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 37, no. 12, 2023, pp. 14 548–14 556.

[19] B. Huang, Z. Wang, J. Yang, J. Ai, Q. Zou, Q. Wang, and D. Ye, "Implicit identity driven deepfake face swapping detection," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2023, pp. 4490–4499.

[20] S. Dong, J. Wang, R. Ji, J. Liang, H. Fan, and Z. Ge, "Implicit identity leakage: The stumbling block to improving deepfake detection generalization," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 3994–4004.

[21] Z. Shang, H. Xie, Z. Zha, L. Yu, Y. Li, and Y. Zhang, "Prrnet: Pixel-region relation network for face forgery detection," *Pattern Recognition*, vol. 116, p. 107950, 2021.

[22] L. Li, J. Bao, T. Zhang, H. Yang, D. Chen, F. Wen, and B. Guo, "Face x-ray for more general face forgery detection," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 5001–5010.

[23] T. S. Gunawan, S. A. M. Hanafiah, M. Kartiwi, N. Ismail, N. F. Za'bah, and A. N. Nordin, "Development of photo forensics algorithm by detecting photoshop manipulation using error level analysis," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 7, no. 1, pp. 131–137, 2017.

[24] H. Liu, X. Li, W. Zhou, Y. Chen, Y. He, H. Xue, W. Zhang, and N. Yu, "Spatial-phase shallow learning: rethinking face forgery detection in frequency domain," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 772–781.

[25] P. Yu, J. Fei, Z. Xia, Z. Zhou, and J. Weng, "Improving generalization by commonality learning in face forgery detection," *IEEE Transactions on Information Forensics and Security*, vol. 17, pp. 547–558, 2022.

[26] 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.

[27] I. Masi, A. Killekar, R. M. Mascarenhas, S. P. Gurudatt, and W. AbdAlmageed, "Two-branch recurrent network for isolating deepfakes in videos," in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part VII 16*. Springer, 2020, pp. 667–684.

[28] C. Miao, Z. Tan, Q. Chu, N. Yu, and G. Guo, "Hierarchical frequency-assisted interactive networks for face manipulation detection," *IEEE Transactions on Information Forensics and Security*, vol. 17, pp. 3008–3021, 2022.

[29] 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.

[30] 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.

[31] 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.

[32] N. Dufour and A. Gully, "Contributing data to deepfake detection research," *Google AI Blog*, vol. 1, no. 2, p. 3, 2019.

[33] M. Tan and Q. Le, "Efficientnet: Rethinking model scaling for convolutional neural networks," in *International conference on machine learning*. PMLR, 2019, pp. 6105–6114.

[34] 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.

[35] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollar, "Focal loss for dense object detection," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 2980–2988.

[36] D. Cozzolino, J. Thies, A. Rössler, C. Riess, M. Nießner, and L. Verdoliva, "Forensictransfer: Weakly-supervised domain adaptation for forgery detection," *arXiv preprint arXiv:1812.02510*, 2018.

[37] D. Li, Y. Yang, Y.-Z. Song, and T. Hospedales, "Learning to generalize: Meta-learning for domain generalization," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 32, no. 1, 2018.

[38] K. Sun, H. Liu, Q. Ye, Y. Gao, J. Liu, L. Shao, and R. Ji, "Domain general face forgery detection by learning to weight," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 35, no. 3, 2021, pp. 2638–2646.

[39] K. Sun, T. Yao, S. Chen, S. Ding, J. Li, and R. Ji, "Dual contrastive learning for general face forgery detection," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 36, no. 2, 2022, pp. 2316–2324.

[40] B. Liu, B. Liu, M. Ding, T. Zhu, and X. Yu, "Ti2net: temporal identity inconsistency network for deepfake detection," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2023, pp. 4691–4700.

[41] C. Miao, Q. Chu, W. Li, S. Li, Z. Tan, W. Zhuang, and N. Yu, "Learning forgery region-aware and id-independent features for face manipulation detection," *IEEE Transactions on Biometrics, Behavior, and Identity Science*, vol. 4, no. 1, pp. 71–84, 2021.

[42] Y. Qian, G. Yin, L. Sheng, Z. Chen, and J. Shao, "Thinking in frequency: Face forgery detection by mining frequency-aware clues," in *European conference on computer vision*. Springer, 2020, pp. 86–103.

[43] 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.

[44] Y. Wang, K. Yu, C. Chen, X. Hu, and S. Peng, “Dynamic graph learning with content-guided spatial-frequency relation reasoning for deepfake detection,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 7278–7287.

[45] J. Fei, Y. Dai, P. Yu, T. Shen, Z. Xia, and J. Weng, “Learning second order local anomaly for general face forgery detection,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 20 270–20 280.

[46] H. Zhao, W. Zhou, D. Chen, T. Wei, W. Zhang, and N. Yu, “Multi-attentional deepfake detection,” in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 2185–2194.

[47] H. Chen, Y. Li, D. Lin, B. Li, and J. Wu, “Watching the big artifacts: Exposing deepfake videos via bi-granularity artifacts,” *Pattern Recognition*, vol. 135, p. 109179, 2023.

[48] C. Tian, Z. Luo, G. Shi, and S. Li, “Frequency-aware attentional feature fusion for deepfake detection,” in *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP).* IEEE, 2023, pp. 1–5.

[49] S. Chen, T. Yao, Y. Chen, S. Ding, J. Li, and R. Ji, “Local relation learning for face forgery detection,” in *Proceedings of the AAAI conference on artificial intelligence*, vol. 35, no. 2, 2021, pp. 1081–1088.