

GLFF: Global and Local Feature Fusion for Face Forgery Detection

Yan Ju, Shan Jia, Jialing Cai, Siwei Lyu, *Fellow, IEEE*,

arXiv:2211.08615v6 [cs.CV] 13 Feb 2023

Abstract—With the rapid development of deep generative models (such as Generative Adversarial Networks and Auto-encoders), AI-synthesized images of the human face are now of such high quality that humans can hardly distinguish them from pristine ones. Although existing detection methods have shown high performance in specific evaluation settings, e.g., on images from seen models or on images without real-world post-processing, they tend to suffer serious performance degradation in real-world scenarios where testing images can be generated by more powerful generation models or combined with various post-processing operations. To address this issue, we propose a Global and Local Feature Fusion (GLFF) framework to learn rich and discriminative representations by combining multi-scale global features from the whole image with refined local features from informative patches for face forgery detection. GLFF fuses information from two branches: the global branch to extract multi-scale semantic features and the local branch to select informative patches for detailed local artifacts extraction. Due to the lack of a face forgery dataset simulating real-world applications for evaluation, we further create a challenging face forgery dataset, named DeepFakeFaceForensics (DF^3), which contains 6 state-of-the-art generation models and a variety of post-processing techniques to approach the real-world scenarios. Experimental results demonstrate the superiority of our method to the state-of-the-art methods on the proposed DF^3 dataset and three other open-source datasets.

Index Terms—Face Forgery Detection, AI-synthesized Faces, Face Forgery Dataset, Image Forensics, Feature Fusion, Attention Mechanism.

I. INTRODUCTION

THE creation of realistic synthetic images/videos using deep learning models has seen rapid development with improved visual qualities and run-time efficiencies in face forgery tasks such as entire face synthesis, identity swap (as known as DeepFakes), attribute manipulation, and expression swap in recent years. Notable examples include high-quality entire face images synthesized with the generative adversarial networks (GANs) [3] and identity-swapped or face-reenacted videos created using Auto-encoders (AEs) [4], [5]. The synthetic images/videos have become challenging for humans to distinguish [6] and eroded the trust in domains such as social media, politics, the military, geospatial intelligence, and cybersecurity. Malicious media generated by deepfake tools are major threats in impersonation and disinformation, and

This work is supported by the US Defense Advanced Research Projects Agency (DARPA) Semantic Forensic (SemaFor) program, under Contract No. HR001120C0123.

Y. Ju, S. Jia, J. Cai, and S. Lyu are with the Department of Computer Science and Engineering, University at Buffalo (UB), State University of New York, NY, USA. E-mail: (yanju, shanjia, jialingc, siweilyu)@buffalo.edu.

Corresponding author: Siwei Lyu.

correspondingly, a slew of detection methods [7], [8] have been developed to expose them.

Most existing detection methods employ deep neural networks to automatically extract discriminative features from face images and then feed them into a binary classifier to discriminate between the real and fake ones [1], [9]–[11]. These detectors have achieved good performance on images from seen generation models, where the testing samples are generated using the same model structure or similar source data as the training samples. However, their performance tends to decrease considerably in open-world applications, where the testing images come from a different domain such as being generated by unknown models using unseen source data [12], [13]. When facing such diverse and unknown testing samples, the features learned by current detection models can hardly conduct robust and accurate detection [13]. To improve the detection robustness and generalization ability, recent studies have explored different deep learning techniques to learn more robust and generic features, such as using limited subsampling architecture with contrastive learning [14], transfer learning [15], incremental learning [16], representation learning [17], data augmentation [1], spectrum analysis [18], and dual-color feature fusion [19].

Although these detectors have achieved promoted performance in detecting unseen generated images, their models are evaluated in specific evaluation settings, i.e., with early generation models and limited post-processing operations. In real-world applications, the detection methods will inevitably face two kinds of challenges that may lead to performance degradation: 1) The quality of fake images has improved with refined global qualities and subtle artifacts in local regions due to the rapid development of generation models; 2) Well-designed post-processing operations are inevitable, such as image compression when laundering on social media, secondary manipulation, or well-designed adversarial noise for anti-forensics [20]. Previous methods with limited feature representations, e.g., only utilizing global spatial features [1], [14], [17], [19] or focusing on local features [21], [22], can hardly extract rich and strong representations that can generalize to powerful generation models or diverse post-processing operations in real-world images. We illustrate this finding in Fig. 1, which shows that the high performance achieved in seen data domain drops significantly in detecting unseen and diverse post-processed images.

To address this issue, we emphasize the importance of combining discriminative and complementary features inspired by the powerful feature learning strategies of previous fusion-based detection works [9], [19], [23]. Instead of purely focus-

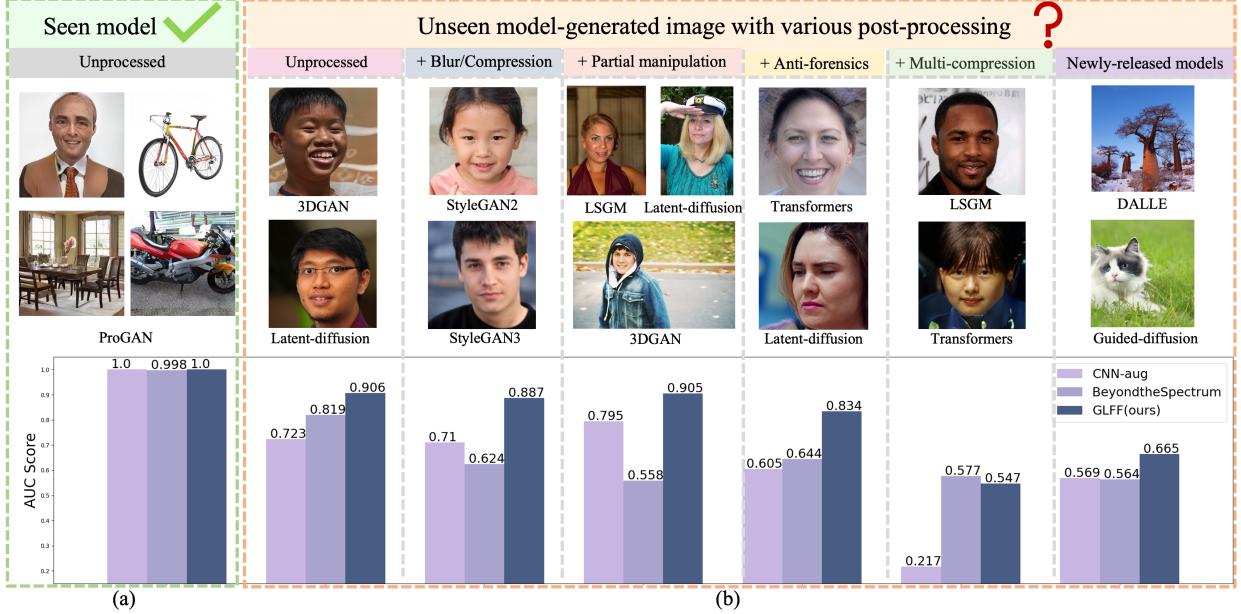


Fig. 1. Illustration of the unseen challenges to current synthetic image detectors. Note that existing state-of-the-art methods CNN-aug [1] and BeyondtheSpectrum [2] achieve good performance on seen unprocessed images but drop greatly on unseen post-processed images, while our method (GLFF) achieves good performance on both seen and unseen, and unprocessed and processed images. See texts for more details. This figure is better viewed in color.

ing on feature fusion from the global domain, we propose a Global and Local Feature Fusion (GLFF) framework to fuse multi-scale global and local subtle features to learn more robust and general feature representation for challenging face forgery detection. Specifically, our GLFF consists of two branches: the global branch combines low-level shallow features with high-level semantic features with a proposed Attention-based Multi-Scale Feature Fusion (AMSFF) module. Taking the fused multi-scale features as guidance, our local branch utilizes the Patch Selection Module (PSM) to locate informative patches automatically and then extract local and subtle artifacts. The two-branch features are finally fused using an attention-based module and fed into a binary classifier to distinguish between real and fake face images.

Considering the lack of a comprehensive face forgery dataset to evaluate how face forgery detection methods will perform in a real-world scenario, we further present a challenging face forgery dataset, DF^3 , which combines state-of-the-art generation models with a variety of post-processing operations to approach the real-world applications. Specifically, we include 6 recent generation models, including GAN-based, Transformer-based, and diffusion-based generation models proposed in recent two years. Furthermore, we consider the following post-processing operations conducted on the generated face images: 1) common operations used on social media, e.g., JPEG compression and Gaussian blurring; 2) secondary manipulation with face blending, which blends the generated face into a real background to enhance the authenticity of the forged images; 3) several well-designed anti-forgery algorithms; 4) a novel multi-image compression to create the comprehensive and diverse dataset; 5) mixed operations by combining multiple operations. Overall, the DF^3 dataset contains approximately 46,400 generated face

images.

Our main contributions can be summarized as follows:

- We propose a novel detection framework (GLFF) to fuse multi-scale global features from the whole image and subtle local features from multiple informative patches. An Attention-based Multi-scale Feature Fusion (AMSFF) module is developed for feature fusion and a novel Patch Selection Module (PSM) is designed for local patch selection.
- We construct a large-scale and highly-diverse face forgery dataset DF^3 by considering 6 state-of-the-art generation models and 5 post-processing operations to approach the real-world applications. To our best knowledge, this is the first dataset considering both advanced generation models and diverse post-processings for challenging face forgery detection.
- Extensive experimental results demonstrate the difficulty of detecting forged faces in the proposed DF^3 dataset for existing detection methods, with a performance degradation of over 30%. Our method achieves outstanding generalization ability on the DF^3 dataset and several public datasets.

A preliminary version of this work was in [12], in which we proposed a two-branch model to combine global information and subtle local features to detect diverse types of AI-synthesized images. This work extends on that in two ways: 1) In the global branch, inspired by recent work [9] revealing that slight artifacts caused by forgery methods tend to be preserved in the shallow features, we extract and fuse low-level features with high-level semantic features for more fine-grained feature learning. Based on the fused global feature, our local branch selects informative patches for the classification. 2) To evaluate the performance of the proposed method

under real-world applications, we construct a challenging face forgery dataset DF^3 as a benchmark by considering 6 state-of-the-art generation models and 5 post-processing operations to simulate the real-world applications.

The remainder of the article is organized as follows. Related works are briefly reviewed in Section II. The proposed GLFF method for face forgery detection is presented in Section III. The DF^3 dataset is introduced in Section IV. Section V shows the experimental results and corresponding analysis. The conclusion and future work are presented in Section VI.

II. RELATED WORK

A. Face Forgery Techniques

Existing face forgery techniques can be classified into four types: entire face synthesis, identity swap (DeepFakes), attribute manipulation, and expression swap. The development of deep learning techniques, especially the Generative Adversarial Networks (GANs) [24], Autoencoders (AEs) [25], and diffusion models [26], has made face manipulation increasingly realistic. GANs are usually used to synthesize non-existent entire images, such as ProGAN [27], StyleGAN-V1/2/3 [3], [28], [29], CycleGAN [30], and Pix2Pix [31]. Recently, powerful diffusion models have been explored and achieved state-of-the-art synthesis results on image generation [26], [32], [33]. For partially manipulated images/videos such as identity swap, attribute manipulation, and expression swap, most methods are based on AE models [4], [5] and GANs [34], [35].

B. Face Forgery Detection

A variety of detection methods have been developed to distinguish between real and synthetic/manipulated images. A common approach is to train deep neural networks as detectors to solve the binary classification task [1], [9]–[11], [36]. These methods only utilize global feature from the entire image and neglect the local artifacts. Another branch of detection methods extracts complementary features such as spectrum analysis [18] and dual-color feature fusion [19].

To obtain more generalized features for diverse manipulation detection, recent works demonstrated the importance of low-level features and features from local patch for detection and localization [9], [10], [22], [23], [37], [38]. Specifically, Zhao et al. [9] observe that the slight artifacts caused by forgery methods tend to be preserved in the high-frequency component of the shallow features. Therefore, they propose a multi-attentional framework to solve deepfake detection by incorporating shallow features. The study in [22] proposes a patch-based classifier and obtain good generalization ability. The method in [37] also extracts features from various facial parts. However, this method utilizes local artifacts without considering the global semantic features, which may not generalize well to more diverse and subtle forgery detection tasks. Most similar to our work is [23], in which the authors propose a two-stream face forgery detection model to learn local subtle artifacts and global spatial features for face forgery detection. However, their local and global features are fused in the score level but our method fuses them in the

feature level. Moreover, their local patches are pre-defined by dividing the image evenly but our method introduces PSM to select the patches automatically. Besides of the feature-learning methods mentioned-above, recent works also utilize various training strategies to improve the generalization, such as transfer learning, incremental learning, contrastive learning, representation learning, and self-supervised learning [14]–[17], [39]–[41], etc. Moreover, [42] proposes a model to learn fingerprints commonly shared by GANs for open-world GAN image detection based on out-of-distribution detection. Data augmentation has been used to improve generalization ability and robustness in [1], [43].

C. Face Image Forgery Datasets

There have been several public image forgery datasets. Karras et al. [28] released 100k-Generated-Images including 100k synthetic face images generated using their proposed StyleGAN [28] trained on FFHQ dataset [28]. 100k-Faces [44] is another dataset containing 100k synthetic images generated using StyleGAN [28] but trained on a more diverse dataset. Dang et al. introduced a Diverse Fake Face Dataset (DFFD) generated using StyleGAN [28] and ProGAN [27] models in [45]. Neves et al. proposed the iFakeFaceDB database [46] using StyleGAN [28] and ProGAN [27]. Different from previous datasets, the iFakeFaceDB dataset is processed with a anti-forgery algorithm GANprintR (GAN Fingerprint Removal) to remove the artifacts and thus confuse fake detectors. Wan et al. [1] collected a diverse dataset created from 11 synthesis models, including GANs, models with perceptual loss, low-level vision models, and DeepFakes to evaluate the generalization ability of their proposed detection method. Recently, Asnani et al. presented a fake image dataset with 116K images generated by 116 different generative models for deepfake detection and model attribution [11].

Existing datasets adopted generative models as the only tool to generate fake images without considering any post-processing operations in real-world applications. Although the OpenForensics dataset [47] utilizes several post-processing operations to blend GAN-generated faces into real images, and iFakeFaceDB dataset [46] considers the anti-forgery application, they only consider limited post-processing operations, which is far different from real-world scenarios. Moreover, state-of-the-art models such as transformer-based or diffusion-based models are not covered in existing datasets. Therefore, impressive performance on these datasets can not ensure that current detection models can work successfully under a more challenging scenario where the manipulated images are generated not only by advanced models but also with well-designed post-processings.

III. GLFF FOR FACE FORGERY DETECTION

To learn generalized representations for face forgery detection, we emphasize the importance of learning rich and complementary features in detecting diverse face forgeries and propose the Global and Local Feature Fusion (GLFF) framework by combining multi-scale global representations with informative local features. In this section, we first introduce

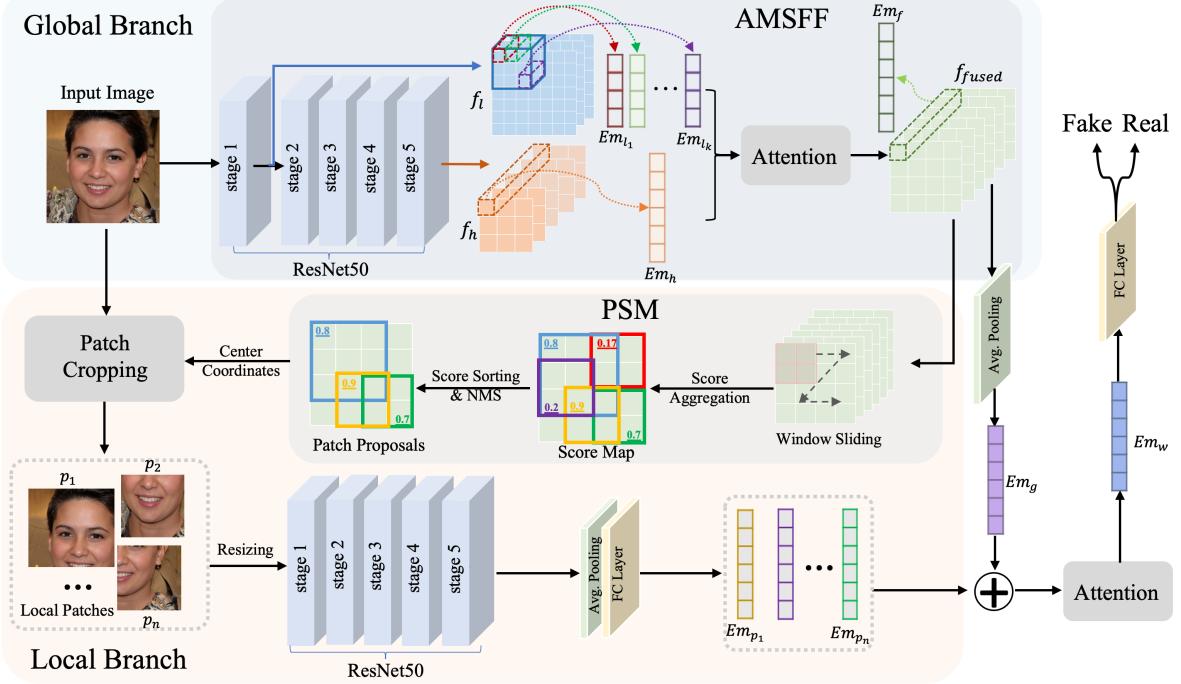


Fig. 2. **The architecture of GLFF.** The global branch uses Attention-based Multi-scale Feature Fusion module (AMSFF) to extract low-level and high-level spatial features, and the local branch extracts subtle feature from local patches selected by Patch Selection Module (PSM). The Attention mechanism is utilized to fuse global and local embeddings for the classification.

the overall architecture of our detection method, and then elaborate on each module of the method, including the global branch with a novel feature fusion scheme, the local branch with a patch selection module, and the final classification module.

A. Overall Architecture

The overall structure of our GLFF model is illustrated in Fig. 2. In the global branch, the input image is fed into the backbone network, which is based on the ResNet-50 [48] in our implementation, considering its concise architecture and competitive performance in various face-related classification tasks [1], [49], [50]. The Attention-based Multi-Scale Feature Fusion (AMSFF) module is designed to aggregate low-level and high-level features extracted from the ResNet-50 model for global representation. To extract informative local features, the fused global representation is then fed into the local branch, from which the proposed Patch Selection Module (PSM) first selects informative patches with higher energy, and thus higher contribution to the classification [51]. The local features are extracted from these selected patches and then fused with the global representation with an attention-based module for the final binary classification.

B. Global Branch

The global branch in our model is designed to learn multi-scale global features and guide local branch patch selection. Inspired by the previous findings that enhancing the feature from shallow layers can prevent the subtle difference from disappearing in the deep layers for face-swapped deepfake

detection [9], we develop an attention module to aggregate hierarchical features to fully explore the subtle artifacts. Different from previous works which combine different features based on concatenation [52] or bilinear pooling [9], we fuse multi-scale features by designing an Attention-based Multi-scale Feature Fusion (AMSFF) module based on the multi-head attention mechanism [53].

Given an input image, the AMSFF module employs the ResNet50 [48] to extract low-level feature maps F_l from the shallow convolutional layer (stage 1 in Fig. 2) and high-level semantic feature maps F_h from the deep convolutional layer (stage 5 in Fig. 2). The stage represents a group of convolutional and pooling layers in ResNet-50 and more details about the layers in each stage are shown in Fig. 3. Assuming the size of F_h be $H_h \times W_h \times C_h$, for each pixel in F_h , the AMSFF aggregates the values along the channel dimension as a vector E_{m_h} , whose size is $1 \times 1 \times C_h$. Assuming that the size of F_l is $H_l \times W_l \times C_l$, we divide F_l into patches evenly without overlap, and the size of each patch is $\frac{H_l}{H_h} \times \frac{W_l}{W_h}$. In the cropped patch, each pixel vector is taken as an embedding $E_{m_{l,i}}$ ($i \in [1, k]$, k is the number of pixels in the cropped patch). To effectively fuse the low-level and high-level features, AMSFF concatenates the k embeddings from low-level feature maps and 1 embedding from high-level feature maps to learn the correlations among these embeddings and combine them as a fused embedding E_{m_f} with the learned weights. Specifically, the concatenated $k + 1$ embeddings are inputted into AMSFF as the query Q, the key K, and the value V. Multi-head attention with s heads is used. The i th attention head h_i ($i \in [1, s]$) is computed with the scaled dot product

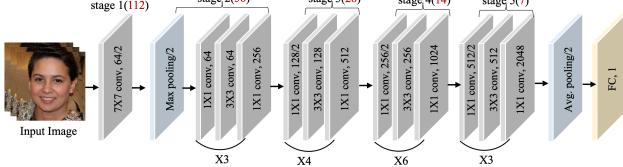


Fig. 3. The architecture of ResNet-50 in our AMSFF module.

attention as following [53]:

$$h_i = \text{softmax}\left(\frac{\mathbf{W}_i^Q \mathbf{Q} \times (\mathbf{W}_i^K \mathbf{K})^T}{\sqrt{d}}\right) \times \mathbf{W}_i^V \mathbf{V} \quad (1)$$

where \mathbf{W}_i^Q , \mathbf{W}_i^K , \mathbf{W}_i^V are parameter matrices and d is the dimension of K . The attention scores of all heads are concatenated as the input of a linear transformation to obtain the value of multi-head attention as follows [53]:

$$E_{m_f} = \text{Concat}(h_i) \mathbf{W}^o \quad (i \in [1, n]) \quad (2)$$

where \mathbf{W}^o is also a parameter matrix. The fused one dimensional vector E_{m_f} is taken as a pixel vector on the fused feature maps. After iterating from all the pixels on the high-level feature maps, the fused feature maps, denoted as $F_{fused} \in \mathbb{R}^{H_h \times W_h \times C_h}$ with C_h channels and $H_h \times W_h$ spatial size. The shape of the fused feature maps is the same as the high-level feature maps, but with more dense information that each pixel contains fused information from a high-level embedding and several low-level embeddings. The global embedding E_{m_g} is finally acquired by feeding F_{fused} into an average pooling layer.

C. Local Branch with Patch Selection Module

Our local branch is designed to select the most n informative patches (see p_1, p_2, \dots, p_n , in Fig. 2) for discriminative local embeddings $E_{m_{p_1}}, E_{m_{p_2}}, \dots, E_{m_{p_n}}$ extraction. Assuming that the regions with higher energy in global fused feature maps F_{fused} can usually provide more information for classification task [51], we introduce the Patch Selection Module (PSM) to select informative patches by calculating the patch score from the fused feature maps F_{fused} .

During the calculation of the patch score, the activation map A is estimated by aggregating F_{fused} along the channel dimension as shown as $A(x, y) = \sum_{j=1}^{C_h} F_{fused}^j(x, y)$. Then the average score of a window on the activation map indicates its “informativeness” on the input image. Specifically, a sliding window with $H_w \times W_w$ spatial size is used to aggregate the information on the activate map A , and the score of each window S_w is calculated as following:

$$S_w = \frac{\sum_{x=0}^{W_w-1} \sum_{y=0}^{H_w-1} A(x, y)}{W_w \times H_w} \quad (3)$$

By sorting the scores from all windows and reducing patch redundancy with non-maximum suppression (NMS), n windows are selected as the patch proposals. The central coordinates of these patch proposals are mapped back to the input image to guide the patch cropping. After patch cropping,

the most n informative local patches, p_1, p_2, \dots, p_n , are fed into the ResNet50 for local embeddings $E_{m_{p_1}}, E_{m_{p_2}}, \dots, E_{m_{p_n}}$ extraction.

D. Feature Fusion for Classification

The extracted global embedding E_{m_g} from the global branch and local embeddings $E_{m_{p_1}}, E_{m_{p_2}}, \dots, E_{m_{p_n}}$ from informative n local patches in the local branch are concatenated and fed into the multi-head attention module for fusion. Following the Equations (1) and (2), the final embedding E_{m_w} is calculated for the final binary classification. We used the Softmax classifier and Binary Cross Entropy loss function in our implementation.

IV. DF³ DATASET

Existing face forgery datasets contain faces generated with limited generation models without considering well-defined post-processings. To construct a highly-diverse face forgery dataset which approaches real-world applications, we create the DF^3 dataset with newly emerged generation models and a variety of post-processing techniques. In this section, we will first introduce basic statistical information about the dataset, and then give details of the dataset construction.

The DF^3 dataset contains approximately 46,400 fake images from 6 generation models and 5 post-processing protocols. The detailed statistics of each protocol are shown in Table I. Examples of the generated faces can be found in Fig. 4.

Our dataset construction consists of two steps: 1) Unprocessed fake image generation; 2) Post-processing implementation. For unprocessed fake image generation, we selected 6 state-of-the-art generative models, i.e., 3 unconditional GANs: StyleGAN2 [3], StyleGAN3 [29], 3DGAN [54], 1 Transformers [55], and 2 diffusion models: LSGM [32] and Latent-diffusion [26]. We use the following two criteria in selecting these models: 1) they show impressive performance on face generation; 2) they all provide open-source pre-trained models which are trained on a large-scale and high-quality face images. Specifically, we generated around 1,000 face images with each model. For the post-processing implementation, different from existing studies which merely consider image compression or blurring that can be easily addressed by data augmentation during model training, we include more diverse post-processing operations to create challenging face forgeries as well as to promote more robust detection methods. Considering that the artifacts in generated face images can be suppressed or destroyed by both common post-processing and well-designed anti-forgery, we propose 5 protocols, including common post-processing (e.g., JPEG Compression and Gaussian Blur caused by image laundering), face blending that merges a fake face to a real background to improve authenticity, anti-forgery with well-designed adversarial attacks, multi-image compression to add complex noises, and mixed version of the previous operations.

TABLE I
DETAILS OF THE PROPOSED DF^3 DATASET.

Generation Model	Protocol-1 Unprocessed	Protocol-2 Common Post-processing	Protocol-3 Face Blending	Protocol-4 Anti-forensics	Protocol-5 Multi-image Compression	Protocol-6 Mixed
StyleGAN2 [3], 2020	1,000	1,000	994	996	1,000	2,652
StyleGAN3 [29], 2021	1,000	1,000	995	996	1,000	2,609
3DGAN [54], 2022	1,000	1,000	984	996	1,000	2,588
Transformers [55], 2021	961	961	959	960	961	2,710
LSGM [32], 2021	1,199	1,199	998	1,191	1,199	2,680
Latent-diffusion [26], 2022	1,000	1,000	998	996	1,000	2,694
Total	6,160	6,160	5,928	6,135	6,160	15,933

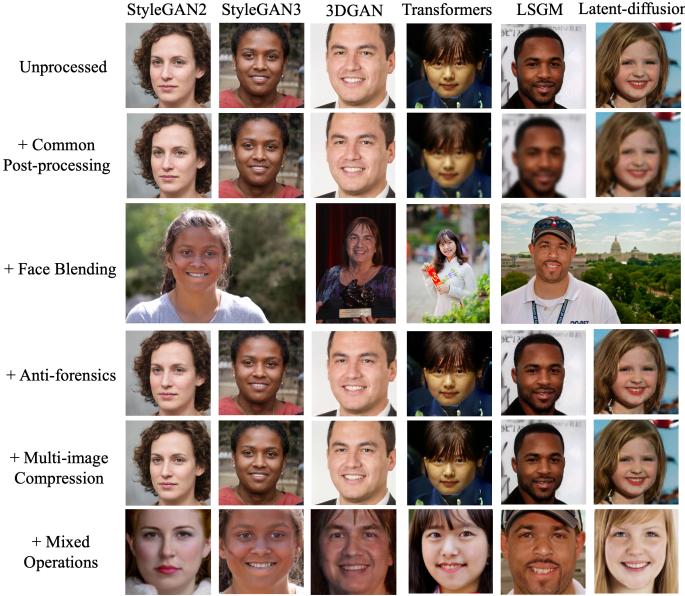


Fig. 4. Examples from the DF^3 dataset (best viewed online in color with zoom-in). From up to down, the original image is generated by a generation model followed by five post-processing operations.

1) *Common Post-processings*: Two popular post-processing methods are included for this protocol: JPEG Compression and Gaussian Blur considering that they are widely used in image laundering by social media platforms to minimize the bandwidth and accelerate the transmission. Given a generated face image, we randomly select one set from three common post-processing operations, i.e., JPEG Compression, Gaussian Blur, and Gaussian Blur combined with JPEG Compression. In JPEG Compression, the quality parameter is discretely selected randomly from [20, 90]. In Gaussian Blur, the input image is blurred with parameter σ continuously selected from [1, 4] randomly. Gaussian Blur combined with JPEG Compression conducts Gaussian Blur followed by JPEG Compression with same parameter selection strategies.

2) *Face Blending*: Current fake image generation models usually generate realistic faces with a large face region but without any semantically meaningful background (as shown in the first row of Fig. 4), which limits the application of fake faces and also becomes an important clue for detection. To improve the diversity and the authenticity of fake faces, we propose to blend the fake faces into various real-world scenes to generate more complex and living photos. In the

face blending subset, we use the SimSwap [56] to swap our fake faces from 6 generation models to the real-world face pictures in FFHQ-unaligned dataset [28], which has a high diversity in face region and semantic background. In total, 5,928 images are generated under this protocol after cleaning the failed samples manually, and more examples can be found in Fig. 4. To the best of our knowledge, this is the first dataset to combine fake faces with real-world scenes. We believe this kind of manipulation is significant while challenging to detect due to the following two reasons: 1) this manipulation improves the authenticity of fake faces and helps expand the application of fake faces (not only used as account profiles); 2) such face-swap manipulation improves the difficulty in detection due to the diverse face regions and complex post-processing from color tuning and blending steps which may weaken the artifacts of the original generated images and introduce new noises.

3) *Anti-forensics*: The design of effective adversarial attacks is another widely studied method to challenge the detection of fake faces. To evaluate how detection methods can perform under this well-designed post-processing, we include two popular algorithms as the anti-fake method to confuse current detectors, including the CW attack algorithm [57] for adversarial examples generation and the GANPrintR [46] for GAN image artifacts removing. Specifically, we adopt the CW algorithm to attack a basic CNN-based detection model [1]. As there are many detection models that are developed from this basic CNN model, we attack this basic CNN detection model to explore if this attack works for other models with similar architecture. Another anti-fake method GANPrintR [46] utilizes an auto-encoder as a low-pass filter to remove more high-frequency components in the GAN-generated images, which has been proven quite effective in misleading the detector to make wrong decisions. We randomly apply either the CW attack algorithm or GANPrintR to the generated face image and generate a total of 6,135 images for this protocol. Fig. 4 shows the high quality of images in this subset.

4) *Multi-image Compression*: Considering that single-image compression can be easily addressed by data augmentation in the training process, we propose a novel multi-image compression strategy inspired by the H.264 video compression, so that we can not only compress the single image in the spatial domain but also compress temporal redundancy of similar blocks between frames sequentially. High-frequency information has been proven to be an important cue for GAN-generated image detection and attribution [11], [18], [58].

We believe that the combination of intra-image compression with cross-image compression will further suppress the high-frequency information in fake images due to the reduction of high redundancy in both spatial and temporal domains. Specifically, we randomly group multiple synthesized images as a video sequence and compress it with H.264 using Hand-Brake [59] tool with the default quality parameter as 22. Then the processed images are obtained by frame extraction to compose the subset with 6,160 images.

5) *Mix Subset*: We finally mix the previous 4 types of post-processing operations to further approach real-world applications where the fake images can experience multiple manipulations to significantly fool the detection tool. Given a generated image, we randomly select several manipulations out of the 4 manipulations and consider the following combinations: Face Blending + Common Post-processing, Face Blending + Multi-image Compression, Multi-image Compression + Anti-forensics, Face Blending + Common Post-processing+Anti-forensics and Face Blending + Multi-image Compression + Anti-forensics. A total of 15,933 images are generated for this protocol.

V. EXPERIMENTS

In this section, we conduct extensive experiments to evaluate the effectiveness of the proposed detection method. We first introduce the implementation details of our method, followed by state-of-the-art baselines for comparison and several evaluation datasets. Then we compare and analyze the experimental results of our method on four datasets. Finally, we present the ablation studies and visualization results.

A. Implementation Details

Our model is implemented with Pytorch [60] and trained with a batch size of 64 and base learning rate of 0.0001. The Adam optimizer and the Binary Cross Entropy loss function are used. ResNet-50 [48], pre-trained with ImageNet, is utilized as the backbone network to extract features from different scales. The input image is resized to $224 \times 224 \times 3$. Before resizing, the input image is blurred with $\sigma \sim \text{Uniform}[0, 3]$ and JPEG-ed with quality $\sim \text{Uniform}\{30, 31, \dots, 100\}$ with 10% probability during training process. Fused feature maps F_{fused} with 2048 channels and 7×7 spatial size are inputted into local branch to guide PSM. In PSM implementation, two sliding windows with the spatial size of 3×3 and 2×2 are used to aggregate the score of each local patch on the activation map. 6 local patches are selected including 3 patches from window size 3×3 and 3 patches from window sizes 2×2 . To utilize patches with different resolution, we map the patches from window size 3×3 into 224×224 while the patches from window sizes 2×2 are mapped into 112×112 on the input image. These 6 patches are further resized into 224×224 and fed into ResNet-50 for local embeddings extraction. The dimension of extracted global embedding and local embedding is set as 128. In AMSFF module and the attention-based global and local feature fusion module, 3 attention layers with 4 heads are utilized to combine all the extracted features.

B. Baseline Methods

We use 5 state-of-the-art generated image detection models as the baselines, of which 3 models are based on spatial information: CNN_aug [1], No_down [10], and PSM [12] and 2 are based on spectral information: GAN-DCT [18] and BeyondtheSpectrum [2]. We follow the settings in their paper and train them on our training dataset from scratch except for No_down, in which the training code was not provided so we used their released pre-trained weights.

For evaluation metrics, we utilize commonly used metrics including Overall Accuracy (OA) and Area Under ROC Curve (AUC) to evaluate our model and the baselines. We also report the Average AUC (Avg.AUC) by averaging it over the 6 protocols in our experiment. Moreover, ROC Curves are shown for a qualitative comparison.

C. Datasets

The training dataset we used for our model training is same as that in [1]. It comprises 362K real images from LSUN dataset [61] with 20 object classes, including person, cat, dining room, etc. 362K images generated with ProGAN [27] are provided as fake images. The spatial size of all training images is 256×256 .

The testing dataset consists of two parts. The first part is from the proposed challenging DF^3 dataset. This dataset is divided into 6 subsets according to different post-manipulations. More details can be found in Table I. The second part comes from existing datasets (i.e., CNN-aug [1] and ReGMs [11]) considering that they contain diverse model-generated images. Moreover, we collect a small dataset of synthetic images generated by 5 newly released generation models, including StyleGAN2-distillation [62], StyleGAN3 [29] and the state-of-the-art diffusion models such as DALLE2 [63], Guided-diffusion [33], and also including partially manipulated images using background matting method [64] to blend the generated images to a real background.

TABLE II
DETAILS OF THE HIGHLY-DIVERSE TESTING DATASETS.

Dataset Source	# Models	# Images(Real/Fake)
DF^3 (ours)	6	46,476 (0/46,476)
CNN-aug [1]	13	90,329 (45,169/45,160)
ReGMs [11]	13	37,709 (24,709/13,000)
Newly-released models	5	12,388 (0/12,388)

Since our DF^3 dataset only consists of fake images, we further collect 1,000 real images as negative samples to enable the metrics calculation such as AUC. These real images are randomly selected from the FFHQ dataset [28] because 1) it consists of high-quality images with considerable variation in terms of age, ethnicity, and image background and 2) it is widely used in several generation models as a training dataset [28]. During testing, these real images are taken as negative class, and images generated by each generation model are taken as the positive class to calculate classification metrics. More details of the testing dataset in the second part are listed in Table II. It is worth noting that our experiments mainly focus on cross-domain testing to evaluate the generalization

TABLE III
EVALUATION RESULTS (AUC) ON THE PROPOSED DF^3 DATASET. THE HIGHEST VALUE IS HIGHLIGHTED IN BLACK.

Test Data	CNN-aug [1]	GAN-DCT [18]	Nodown [10]	BeyondtheSpectrum [2]	PSM [12]	GLFF(Ours)
Unprocessed	0.723	0.656	0.970	0.819	0.901	0.906
Common Post-processing	0.710	0.443	0.823	0.624	0.878	0.887
Face Blending	0.795	0.483	0.888	0.558	0.877	0.905
Anti-forensics	0.605	0.504	0.894	0.644	0.863	0.834
Multi-image Compression	0.217	0.646	0.105	0.577	0.411	0.547
Mixed	0.528	0.497	0.468	0.470	0.724	0.801
Average	0.596	0.538	0.691	0.616	0.775	0.813

ability of different detection methods (as previous studies do [1], [10]).

D. Evaluation on DF^3 dataset

Table III reports the AUC scores on each subset of the proposed DF^3 dataset. To be noted, the AUC of each subset is the average AUC among 6 generation models. The average AUC among all 6 subsets is also listed. All the methods achieve high accuracy on the unprocessed image subset because these images are directly generated with the 6 generation models without any post-processing so the fingerprints can be spotted by the current detectors. Our method achieves an AUC of 0.906, slightly worse than the Nodown [10] method, which argues that avoiding down-sampling layers could save more artifacts to help the forgery detection.

Compared with the performance on the unprocessed image, it can be seen that the performance of CNN-aug, PSM, and our model with data augmentation during training drops slightly while greatly on others on the common post-processing subset. This is because these models adopt JPEG Compression and Gaussian Blur to augment their data during model training.

The performance of GAN-DCT, Nodown, and BeyondtheSpectrum drops greatly on the Face blending subset. This is because the face blending process needs to adjust the color of original generated image and thus remove some low-level generation artifacts, which makes it harder to detect. However, PSM and our model drop slightly on this subset because these two methods exploit details from local patches such that they are more robust to such partial manipulation.

The performance of all methods on the anti-forensics subset drops greatly compared with that of clean images because these anti-forensics algorithms add some noise or remove the artifacts from the original generated image and thus make it more challenging to detect. As the CW attack algorithm is utilized targeted at the CNN-aug detector, so we did not compare CNN-aug with other methods for a fair comparison. For other methods, our model is slightly worse than Nodown, which achieves the best performance on the Anti-forensics subset due to more artifacts saved by avoiding the down-sampling processing.

As for the multi-image compression subset, we utilize video compression as a new compression strategy because of its high compression rate. It can be seen that this processing decreases the performance heavily of all methods except GAN-DCT. This compression method could find similar patches across a group of images and compress them together instead of compressing a single image such that more high-frequency

artifacts in the generated image are removed. However, for GAN-DCT, it exploits DCT to extract spectral features for detection, which may be less influenced by the compression method than other spatial-based methods.

Finally, we test all the methods on the mixed subset, which combine 2 or 3 above-mentioned processings randomly. It can be seen that all the methods achieve poor performance on this subset, which proves that our proposed mixed subset is very challenging and can be utilized as a tool to evaluate the robustness of future detection methods. We also report the average AUC score of each method by averaging it across all the subsets. It can be seen that our model achieves the best performance in terms of Average AUC and AUC on most subsets. This demonstrates the better generalization ability and robustness of our model than other methods, which only use whole image information or only focus on features from small patches without considering multi-scale information combination.

To further demonstrate the superiority of our method qualitatively, we report the ROC curves on 6 generation models from 6 subsets in our DF^3 dataset. We compare our method with the Nodown model [10] and show the results in Fig. 5. It can be observed that the performances on diffusion-based models including the LSGM and Latent-diffusion models and Transformers are lower than that on StyleGAN2, StyleGAN3 and 3DGAN models. This can be attributed to the fact that our detector is trained on images generated by a single GAN model and it can be extended to more general GAN models-generated image detection. However, it is challenging for the detectors to generalize well to the diffusion models/transfomers due to the big gap of model structure and training strategy between GAN models and Transformer/diffusion models. Similar results can be found in the baseline Nodown model. Overall, the results demonstrate that although the performance of our model is slightly lower than the Nodown model for unprocessed and anti-forensics images, our model achieves better performance on the other four protocols, which is consistent with that in Table III.

E. Evaluation on Existing Datasets

There are several existing datasets that consist of fake images from diverse generation models. In this section, we collect them to evaluate the generalization ability of the baselines and our method. Table IV reports the average AUC on existing datasets and our collected dataset. More details of these datasets can be found in Table II. It can be seen that most methods achieve good performance on the testing

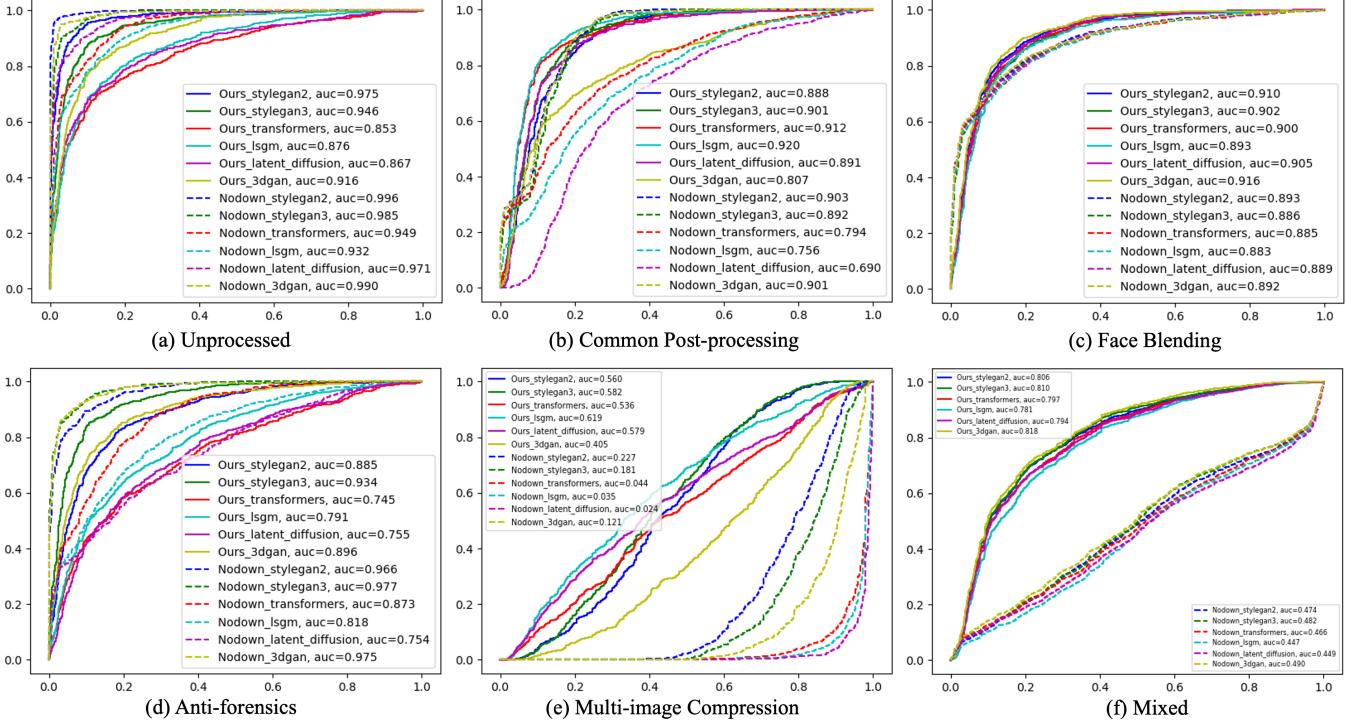


Fig. 5. Comparison of ROC curves under 6 generation models and 6 evaluation protocols in DF^3 dataset.

TABLE IV
EVALUATION RESULTS (AUC) ON PUBLIC DATASETS. THE HIGHEST VALUE IS HIGHLIGHTED IN BLACK.

Dataset	CNN-aug [1]	GAN-DCT [18]	Nodown [10]	BeyondtheSpectrum [2]	PSM [12]	GLFF (Ours)
CNN-aug [1]	0.936	0.603	0.968	0.875	0.942	0.946
ReGMs [11]	0.920	0.589	0.952	0.785	0.967	0.969
Newly-released models	0.569	0.646	0.653	0.564	0.641	0.665

dataset from the CNN-aug [1]. Our method obtains an AUC of 0.946. The testing data from ReGMs [11] consists of more diverse generation models such that the performance on this dataset is slightly worse than that in CNN-aug for most detection methods. Our method achieves the best performance on this dataset with the help of combining global and local information. At last, to evaluate if current detection models are robust to the data generated by newly released models, we test all the methods on our collected dataset and report the AUC in the last row. It can be seen that all the methods drop greatly on this dataset, which demonstrates that the development of the generation model has made existing detectors more and more vulnerable, leading to more wrong predictions. Overall, our GLFF model achieves the best performance on most testing datasets. This demonstrates the outstanding generalization ability of our model benefiting from the fusing of global and local features.

F. Ablation Study

To understand how different model architectures and module settings contribute to the performance of our proposed GLFF detection method, we compare our framework with various variants in terms of different model architectures, different module settings, and module parameters. We evaluate the

variants and our method on the Protocol6-Mixed subset in our proposed DF^3 dataset considering that this subset can represent the general robustness to different manipulations.

1) *Architecture Variants*: In our work, we introduce a global and local feature fusion framework for face forgery detection, in which the AMSFF and PSM modules are proposed to fuse multi-scale features and select patches, respectively. Here, we experiment with several alternative counterparts, i.e., (1) Global Branch Only: only utilizing the global branch in our two-branch model to detect forgery; (2) Local Branch Only: only the local embeddings are utilized for the final classification without being fused with global embedding Em_g ; (3) Global + Local Branch w/o AMSFF: using the same two-branch structure as our GLFF, but removing the AMSFF module to fuse low-level and high-level features; (4) Global + Local Branch w/o PSM: using the same two-branch structure as our GLFF, but selecting patches randomly instead of using PSM. We present the OA and AUC in Table V. It can be seen that only using the global branch achieves the worst performance while our GLFF model is the best in terms of both OA and AUC. This demonstrates that combining global and local features based on our proposed AMSFF and PSM can improve the generalization ability and robustness of our detection method.

TABLE V
PERFORMANCE OF OUR METHOD BASED ON DIFFERENT MODEL ARCHITECTURE.

Model Structure	OA	AUC
Global Branch Only	0.306	0.709
Local Branch Only	0.336	0.746
Global + Local Branch w/o PSM	0.353	0.639
Global + Local Branch w/o AMSFF	0.313	0.724
Ours	0.355	0.801

2) *Determination of Layers in AMSFF*: In the global branch of GLFF, we introduce the AMSFF module to combine features from shallow and deep layers based on attention mechanism for feature extraction and patch selection. To understand how different layer combinations in AMSFF will contribute to the performance of our method, we choose different layer combinations in the AMSFF. There are 5 main stages in the structure of the ResNet-50 model, which are denoted from stage 1-5, respectively, as shown in Fig. 3. As mentioned in [9], subtle artifacts tend to be preserved in shallow layers of the network, thus we choose stage 1 and stage 2 as the candidates of the shallow layer. Conversely, we expect the detection method to grasp different regions of the input with the help of high-level semantic information. Therefore, we use stage 4 and stage 5 as the candidates of the deep layer. Experiment results in Table VI demonstrate that the model achieves the best performance when using stage 1 as the shallow layer and stage 5 as the deep layer.

TABLE VI
PERFORMANCE OF OUR METHOD BASED ON DIFFERENT COMBINATION OF SHALLOW LAYER AND DEEP LAYER IN AMSFF.

Candidates of Layers in AMSFF		Metric	
Shallow Layer	Deep Layer	OA	AUC
stage 1	stage 4	0.298	0.759
stage 1	stage 5	0.355	0.801
stage 2	stage 4	0.330	0.754
stage 2	stage 5	0.332	0.693

3) *Settings in PSM*: In our local branch of the proposed GLFF, we design the PSM module to select distinctive patches on the feature maps outputted from the global branch automatically. This module scans the feature maps with the window and selects the patches with higher energy, which contain more information for classification. During our implementation, the size and the number of windows are pre-defined. To understand how the window size and number of windows in PSM will contribute to the performance of our method, we compare the results in Table VII. Since the combination of stages 1 and 5 achieves the best performance in AMSFF, the size of the fused feature map for patch selection is set as 7×7 . Therefore, we experiment with several reasonable window sizes, i.e., 5×5 , 3×3 and 2×2 and a different number of windows, i.e., 6 (3 per size), 4 (2 per size) and 2 (1 per size). Since more selected windows will lead to more computational resources, we set the maximum number of windows as 6 in our implementation. The experiment results in Table VII show that the model reaches the best performance when scanning

the feature maps with 3 windows with a size of 3×3 and 3 windows with a size of 2×2 in PSM.

TABLE VII
PERFORMANCE OF OUR METHOD BASED ON DIFFERENT SIZE AND NUMBER OF WINDOW IN PSM.

Window Size	# Window	OA	AUC
$5 \times 5, 3 \times 3$	6 (3,3)	0.330	0.779
$3 \times 3, 2 \times 2$	6 (3,3)	0.355	0.801
$3 \times 3, 2 \times 2$	4 (2,2)	0.312	0.766
$3 \times 3, 2 \times 2$	2 (1,1)	0.346	0.749

G. Selected Patch Visualization

To fully understand what patches Patch Selection Module (PSM) focuses on, we visualize the selected patches on both fused feature maps F_{fused} and the corresponding input RGB images in Fig. 6. We show the patches on the images generated by StyleGAN2 and StyleGAN3. Note that the fused feature maps are averaged over the channel dimension and turned to the gray-scale image to show. The more yellow the color is, the stronger energy the region will show. As shown in Fig. 6, when the selected patches on the fused feature map are mapped back to the input RGB images, the RGB patches show some semantic artifacts, such as the weird background artifacts, skin, hair, eyebrow or wrinkles regions with rich texture information.

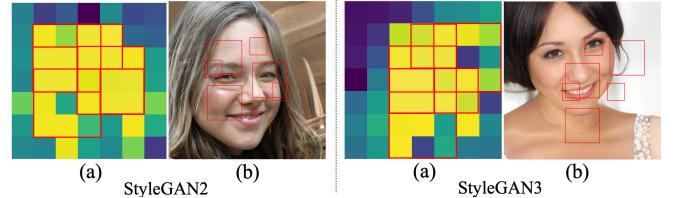


Fig. 6. Visualization of selected patches on (a) Fused feature map, (b) Input RGB image.

VI. CONCLUSION

In this work, we propose a two-branch model GLFF that fuses multi-scale global features and informative local features for face forgery detection. The global branch learns the multi-scale global feature using the proposed AMSFF module, based on which the local branch selects the informative patches using the proposed PSM module and extracts local features. The multi-head attention mechanism is finally used to fuse these complementary features for binary classification. Our model learns to focus on the most discriminative local patches with the help of the global high-level features in an unsupervised way, which allows us to fuse the local and global information more efficiently. To evaluate the performance of our model in the real-world scenario, we also develop a new dataset, DF^3 , which combines several state-of-the-art generation models and various post-processings practiced in real-world applications. To our best knowledge, this is the first face forgery dataset considering both generation models and rich post-processings. Extensive experiments on the proposed DF^3 and existing

datasets demonstrate that the proposed model GLFF outperforms baselines and achieves high accuracy and generalization ability. Both the code and dataset details will be released at <https://github.com/littlejuyan/GLFF>.

In future work, we will first investigate a differentiable patch selection module so that the size and number of local patches can be guided automatically during model training. Moreover, with the proficiency of generation models, we plan to enlarge our dataset by incorporating more advanced model-generated images and post-processing techniques in social media. Besides, developing a model with explainable performance is also worth to be explored. We plan to design a patch selection module with the help of more prior knowledge constraints and explain the selection results more semantically.

REFERENCES

- [1] S.-Y. Wang, O. Wang, R. Zhang, A. Owens, and A. A. Efros, “Cnn-generated images are surprisingly easy to spot... for now,” in *CVPR*, 2020, pp. 8695–8704.
- [2] Y. He, N. Yu, M. Keuper, and M. Fritz, “Beyond the spectrum: Detecting deepfakes via re-synthesis,” in *30th International Joint Conference on Artificial Intelligence (IJCAI)*, 2021.
- [3] T. Karras, S. Laine, M. Aittala, J. Hellsten, J. Lehtinen, and T. Aila, “Analyzing and improving the image quality of stylegan,” in *CVPR*, 2020, pp. 8110–8119.
- [4] Y. Li, X. Yang, P. Sun, H. Qi, and S. Lyu, “Celeb-df: A large-scale challenging dataset for deepfake forensics,” in *CVPR*, 2020, pp. 3207–3216.
- [5] S. Jia, X. Li, and S. Lyu, “Model attribution of face-swap deepfake videos,” *arXiv preprint arXiv:2202.12951*, 2022.
- [6] P. Korshunov and S. Marcel, “Deepfake detection: humans vs. machines,” *arXiv preprint arXiv:2009.03155*, 2020.
- [7] S. Lyu, “Deepfake detection: Current challenges and next steps,” in *2020 IEEE international conference on multimedia & expo workshops (ICMEW)*. IEEE, 2020, pp. 1–6.
- [8] X. Wang, H. Guo, S. Hu, M.-C. Chang, and S. Lyu, “Gan-generated faces detection: A survey and new perspectives,” *arXiv preprint arXiv:2202.07145*, 2022.
- [9] H. Zhao, W. Zhou, D. Chen, T. Wei, W. Zhang, and N. Yu, “Multi-attentional deepfake detection,” in *CVPR*, 2021, pp. 2185–2194.
- [10] D. Gragnaniello, D. Cozzolino, F. Marra, G. Poggi, and L. Verdoliva, “Are gan generated images easy to detect? a critical analysis of the state-of-the-art,” in *ICME*. IEEE, 2021, pp. 1–6.
- [11] V. Asnani, X. Yin, T. Hassner, and X. Liu, “Reverse engineering of generative models: Inferring model hyperparameters from generated images,” *arXiv preprint arXiv:2106.07873*, 2021.
- [12] Y. Ju, S. Jia, L. Ke, H. Xue, K. Nagano, and S. Lyu, “Fusing global and local features for generalized ai-synthesized image detection,” *arXiv preprint arXiv:2203.13964*, 2022.
- [13] N. Hulzebosch, S. Ibrahim, and M. Worring, “Detecting cnn-generated facial images in real-world scenarios,” in *CVPRW*, 2020, pp. 642–643.
- [14] D. Cozzolino, D. Gragnaniello, G. Poggi, and L. Verdoliva, “Towards universal gan image detection,” in *VCIP*. IEEE, 2021, pp. 1–5.
- [15] H. Jeon, Y. Bang, J. Kim, and S. S. Woo, “T-gd: Transferable gan-generated images detection framework,” *arXiv preprint arXiv:2008.04115*, 2020.
- [16] F. Marra, C. Saltori, G. Boato, and L. Verdoliva, “Incremental learning for the detection and classification of gan-generated images,” in *WIFS*. IEEE, 2019, pp. 1–6.
- [17] M. Kim, S. Tariq, and S. S. Woo, “Fretal: Generalizing deepfake detection using knowledge distillation and representation learning,” in *CVPR*, 2021, pp. 1001–1012.
- [18] J. Frank, T. Eisenhofer, L. Schönherr, A. Fischer, D. Kolossa, and T. Holz, “Leveraging frequency analysis for deep fake image recognition,” in *International conference on machine learning*. PMLR, 2020, pp. 3247–3258.
- [19] B. Chen, X. Liu, Y. Zheng, G. Zhao, and Y.-Q. Shi, “A robust gan-generated face detection method based on dual-color spaces and an improved xception,” *IEEE Transactions on Circuits and Systems for Video Technology*, 2021.
- [20] F. Ding, G. Zhu, Y. Li, X. Zhang, P. K. Atrey, and S. Lyu, “Anti-forensics for face swapping videos via adversarial training,” *IEEE Transactions on Multimedia*, 2021.
- [21] D. Cozzolino, G. Poggi, and L. Verdoliva, “Recasting residual-based local descriptors as convolutional neural networks: an application to image forgery detection,” in *Proceedings of the 5th ACM workshop on information hiding and multimedia security*, 2017, pp. 159–164.
- [22] L. Chai, D. Bau, S.-N. Lim, and P. Isola, “What makes fake images detectable? understanding properties that generalize,” in *ECCV*. Springer, 2020, pp. 103–120.
- [23] X. Zhang, S. Wang, C. Liu, M. Zhang, X. Liu, and H. Xie, “Thinking in patch: Towards generalizable forgery detection with patch transformation,” in *Pacific Rim International Conference on Artificial Intelligence*. Springer, 2021, pp. 337–352.
- [24] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, “Generative adversarial networks,” *Communications of the ACM*, vol. 63, no. 11, pp. 139–144, 2020.
- [25] D. P. Kingma and M. Welling, “Auto-encoding variational bayes,” *arXiv preprint arXiv:1312.6114*, 2013.
- [26] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, “High-resolution image synthesis with latent diffusion models,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 10 684–10 695.
- [27] T. Karras, T. Aila, S. Laine, and J. Lehtinen, “Progressive growing of gans for improved quality, stability, and variation,” *arXiv preprint arXiv:1710.10196*, 2017.
- [28] T. Karras, S. Laine, and T. Aila, “A style-based generator architecture for generative adversarial networks,” in *CVPR*, 2019, pp. 4401–4410.
- [29] T. Karras, M. Aittala, S. Laine, E. Härkönen, J. Hellsten, J. Lehtinen, and T. Aila, “Alias-free generative adversarial networks,” *NeurIPS*, vol. 34, 2021.
- [30] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, “Unpaired image-to-image translation using cycle-consistent adversarial networks,” in *ICCV*, 2017, pp. 2223–2232.
- [31] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, “Image-to-image translation with conditional adversarial networks,” in *CVPR*, 2017, pp. 1125–1134.
- [32] A. Vahdat, K. Kreis, and J. Kautz, “Score-based generative modeling in latent space,” *Advances in Neural Information Processing Systems*, vol. 34, pp. 11 287–11 302, 2021.
- [33] P. Dhariwal and A. Nichol, “Diffusion models beat gans on image synthesis,” *Advances in Neural Information Processing Systems*, vol. 34, pp. 8780–8794, 2021.
- [34] Y. Choi, M. Choi, M. Kim, J.-W. Ha, S. Kim, and J. Choo, “Stargan: Unified generative adversarial networks for multi-domain image-to-image translation,” in *CVPR*, 2018, pp. 8789–8797.
- [35] M. Liu, Y. Ding, M. Xia, X. Liu, E. Ding, W. Zuo, and S. Wen, “Stgan: A unified selective transfer network for arbitrary image attribute editing,” in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 3673–3682.
- [36] H. Guo, S. Hu, X. Wang, M.-C. Chang, and S. Lyu, “Robust attentive deep neural network for exposing gan-generated faces,” *arXiv preprint arXiv:2109.02167*, 2021.
- [37] S. Schwarcz and R. Chellappa, “Finding facial forgery artifacts with parts-based detectors,” in *CVPR*, 2021, pp. 933–942.
- [38] M. Yu, S. Ju, J. Zhang, S. Li, J. Lei, and X. Li, “Patch-ddf: Patch-based end-to-end deepfake discriminator,” *Neurocomputing*, 2022.
- [39] 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.
- [40] J. Hu, S. Wang, and X. Li, “Improving the generalization ability of deepfake detection via disentangled representation learning,” in *2021 IEEE International Conference on Image Processing (ICIP)*. IEEE, 2021, pp. 3577–3581.
- [41] Y. Jeong, D. Kim, P. Kim, Y. Ro, and J. Choi, “Self-supervised gan detector,” *arXiv preprint arXiv:2111.06575*, 2021.
- [42] S. Girish, S. Suri, S. S. Rambhatla, and A. Shrivastava, “Towards discovery and attribution of open-world gan generated images,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 14 094–14 103.
- [43] X. Xuan, B. Peng, W. Wang, and J. Dong, “On the generalization of gan image forensics,” in *Chinese conference on biometric recognition*. Springer, 2019, pp. 134–141.
- [44] 100K-Faces. (2018) 100,000 Faces Generated by AI. [Online]. Available: <https://generated.photos/>
- [45] H. Dang, F. Liu, J. Stehouwer, X. Liu, and A. K. Jain, “On the detection of digital face manipulation,” in *Proceedings of the IEEE/CVF*

- Conference on Computer Vision and Pattern recognition*, 2020, pp. 5781–5790.
- [46] J. C. Neves, R. Tolosana, R. Vera-Rodriguez, V. Lopes, H. Proen  a, and J. Fierrez, “Ganprintr: Improved fakes and evaluation of the state of the art in face manipulation detection,” *IEEE Journal of Selected Topics in Signal Processing*, vol. 14, no. 5, pp. 1038–1048, 2020.
- [47] T.-N. Le, H. H. Nguyen, J. Yamagishi, and I. Echizen, “Openforensics: Large-scale challenging dataset for multi-face forgery detection and segmentation in-the-wild,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 10 117–10 127.
- [48] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.
- [49] S. Jia, X. Li, C. Hu, G. Guo, and Z. Xu, “3d face anti-spoofing with factorized bilinear coding,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 31, no. 10, pp. 4031–4045, 2020.
- [50] Y. Li and S. Lyu, “Exposing deepfake videos by detecting face warping artifacts,” *arXiv preprint arXiv:1811.00656*, 2018.
- [51] F. Zhang, M. Li, G. Zhai, and Y. Liu, “Multi-branch and multi-scale attention learning for fine-grained visual categorization,” in *ICMM*. Springer, 2021, pp. 136–147.
- [52] J. Liu, W. Sun, Y. Su, P. Jing, and X. Yang, “Be-calf: bit-depth enhancement by concatenating all level features of dnn,” *IEEE Transactions on Image Processing*, vol. 28, no. 10, pp. 4926–4940, 2019.
- [53] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, “Attention is all you need,” *Advances in neural information processing systems*, vol. 30, 2017.
- [54] E. R. Chan, C. Z. Lin, M. A. Chan, K. Nagano, B. Pan, S. De Mello, O. Gallo, L. J. Guibas, J. Tremblay, S. Khamis *et al.*, “Efficient geometry-aware 3d generative adversarial networks,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 16 123–16 133.
- [55] P. Esser, R. Rombach, and B. Ommer, “Taming transformers for high-resolution image synthesis,” in *CVPR*, 2021, pp. 12 873–12 883.
- [56] R. Chen, X. Chen, B. Ni, and Y. Ge, “Simswap: An efficient framework for high fidelity face swapping,” in *MM '20: The 28th ACM International Conference on Multimedia*, 2020.
- [57] N. Carlini and D. Wagner, “Towards evaluating the robustness of neural networks,” in *2017 ieee symposium on security and privacy (sp)*. Ieee, 2017, pp. 39–57.
- [58] X. Zhang, S. Karaman, and S.-F. Chang, “Detecting and simulating artifacts in gan fake images,” in *2019 IEEE international workshop on information forensics and security (WIFS)*. IEEE, 2019, pp. 1–6.
- [59] HandBrake. (2019) HandBrake. [Online]. Available: <https://github.com/HandBrake/HandBrake>
- [60] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga *et al.*, “Pytorch: An imperative style, high-performance deep learning library,” *NeurIPS*, vol. 32, 2019.
- [61] F. Yu, A. Seff, Y. Zhang, S. Song, T. Funkhouser, and J. Xiao, “Lsun: Construction of a large-scale image dataset using deep learning with humans in the loop,” *arXiv preprint arXiv:1506.03365*, 2015.
- [62] Y. Viazovetskyi, V. Ivashkin, and E. Kashin, “Stylegan2 distillation for feed-forward image manipulation,” in *European conference on computer vision*. Springer, 2020, pp. 170–186.
- [63] A. Ramesh, P. Dhariwal, A. Nichol, C. Chu, and M. Chen, “Hierarchical text-conditional image generation with clip latents,” *arXiv preprint arXiv:2204.06125*, 2022.
- [64] Z. Ke, J. Sun, K. Li, Q. Yan, and R. W. Lau, “Modnet: Real-time trimap-free portrait matting via objective decomposition,” in *AAAI*, 2022.