Hindawi
Security and Communication Networks
Volume 2021, Article ID 5524930, 13 pages
https://doi.org/10.1155/2021/5524930

Research Article

Channel-Wise Spatiotemporal Aggregation Technology for Face Video Forensics

Yujiang Lu,[1] **Yaju Liu**,[2] **Jianwei Fei**,[2] **and Zhihua Xia**[3]

[1]*Changwang School of Honors, Nanjing University of Information Science Technology, Nanjing 210044, China*
[2]*School of Computer and Software, Nanjing University of Information Science Technology, Nanjing 210044, China*
[3]*Engineering Research Center of Digital Forensics, Ministry of Education, School of Computer and Software, Jiangsu Engineering Center of Network Monitoring, Jiangsu Collaborative Innovation Center on Atmospheric Environment and Equipment Technology, Nanjing University of Information Science Technology, Nanjing 210044, China*

Correspondence should be addressed to Zhihua Xia; xia_zhihua@163.com

Received 27 February 2021; Revised 30 June 2021; Accepted 14 August 2021; Published 29 August 2021

Academic Editor: Guoying Zhao

Recent progress in deep learning, in particular the generative models, makes it easier to synthesize sophisticated forged faces in videos, leading to severe threats on social media about personal privacy and reputation. It is therefore highly necessary to develop forensics approaches to distinguish those forged videos from the authentic. Existing works are absorbed in exploring frame-level cues but insufficient in leveraging affluent temporal information. Although some approaches identify forgeries from the perspective of motion inconsistency, there is so far not a promising spatiotemporal feature fusion strategy. Towards this end, we propose the Channel-Wise Spatiotemporal Aggregation (CWSA) module to fuse deep features of continuous video frames without any recurrent units. Our approach starts by cropping the face region with some background remained, which transforms the learning objective from manipulations to the difference between pristine and manipulated pixels. A deep convolutional neural network (CNN) with *skip connections* that are conducive to the preservation of detection-helpful low-level features is then utilized to extract frame-level features. The CWSA module finally makes the real or fake decision by aggregating deep features of the frame sequence. Evaluation against a list of large facial video manipulation benchmarks has illustrated its effectiveness. On all three datasets, FaceForensics++, Celeb-DF, and DeepFake Detection Challenge Preview, the proposed approach outperforms the state-of-the-art methods with significant advantages.

1. Introduction

The rapid development of social networks and the emergence of various mobile applications have promoted the creation and dissemination of digital videos. These videos generally contain rich contents of individuals with regard to face and voice, which are very significant biological information for identity authentication. However, manipulation of these videos will seriously undermine their authenticity. Due to the ever-developing artificial intelligence technologies, existing tools make manipulation easier than ever and more imperceptible. Meantime, the convenient creation and spread of multimedia contents make it uncomplicated for an attacker to obtain their desired material and carry out malicious purposes by these tools. This has become a potential threat to ethics, law, and personal privacy and raised a great alarm. It is therefore of great practical significance to study effective forensics technologies to distinguish these fake videos. However, facial manipulation did not attract too much attention before because the conventional digital image editing methods are easy to spot by naked eyes, and the forensics technologies have been at an advantage until the appearance of deep learning based forgery technologies.

However, in recent years, deep learning based face synthesis, manipulation, and swap technologies which are generally referred to as the term DeepFakes have brought new challenges to face forensics. The original DeepFakes can only swap two faces using a pair of autoencoders that share

the same encoder but is composed of different decoders. They are trained to reconstruct the source and target face images, respectively. Once trained, the target decoder can generate a realistic face image of target identity with the expressions of the source face by being fed with the source face representation output from the source encoder.

Original DeepFakes always produces obvious artifacts when warping faces back to the target images, and this defect has been utilized by the existing approach [1]. In recent years, the continuous development of generative networks can generate very photo-realistic fake faces or completely synthesize videos from a single image and even from portrait paintings [2]. This puts forward higher requirements for forensics approaches in terms of detection accuracy and generalization ability. The forensics approaches have also been developing with the help of deep learning and previous work in digital forensics. According to the clues used, the detection approaches of face video manipulation can be mainly divided into two: intraframe information based and interframe information based. The former focuses on spatial artifacts and realizes video manipulation detection by processing independent frames. The latter captures the dynamic flaws in videos through temporal models like Recurrent Neural Network (RNN) [3] or optical flow [4].

In this paper, we adopt a novel approach to capture the interframe cues by aggregating deep feature sequences channel wisely. It achieves better performance with relatively few parameters. The main contributions of this paper are summarized as follows:

A novel module CWSA is proposed to exploit temporal information by aggregating deep features of consecutive frames but different channels. With a powerful feature extraction backbone EfficientNet B0 [5], our approach reaches the state-of-the-art level on three large datasets.

It is revealed that by keeping the moderate background in face cropping preprocessing, models can learn the difference between pristine and manipulated pixels to obtain gains in detection accuracy.

We demonstrate that *skip connection* preserves the detection-helpful low-level features well. Thus it plays a central role when deep models are used for extracting frame-level features.

This paper is organized as follows. In Section 2, we briefly introduce the existing forensics approaches. In Section 3, we give a detailed description of our approach. The experimental results and analysis are presented in Section 4, and we make a conclusion and prospects for future work in Section 5.

2. Related Work

2.1. Manipulation Forensics. Before the emergence of deep learning based forgery technologies, conventional multimedia contents manipulation such as removal, copy-move, and splicing were realized with image editing technologies. The research of multimedia forensics has been committed to solving the problems of detecting this kind of manipulation for long. These manipulations tend to leave obvious clues, particularly in statistical characteristics caused by editing or compression. Considering this, Cozzolino et al. have proposed a feature-based splicing detection method. Their algorithm computes local features from the cooccurrence matrix of the image residuals, and the parameters extracted from different images were proved to be efficacious on both detection and localization [6]. Similarly, the study in [7] discovered the influence of times of JPEG compression that images go through. With the help of the Nonnegative Matrix Factorization model and histograms of Discrete Cosine Transform, multiple JPEG compression can be successfully detected and indirectly, the authenticity of images.

Another kind of popular approach is to discover clues that are related to the camera itself. In 2006, Lukas et al. proposed to identity camera models through photoresponse nonuniformity, a pattern that reveals the different sensitivity of pixels to light caused by the inhomogeneity of silicon wafers [8]. Researchers also found out camera-related patterns left in out-camera processing history. In [9], Cozzolino et al. have researched to detect and localize forgeries by a camera-based noise pattern. This noise pattern is produced during the compression or gamma correction and can be seen as unique fingerprints of specific camera models. However, the estimation of this noise requires a considerable number of samples, and when encountered with an unknown camera model, detection approaches based on noise pattern would show weakness.

2.2. GAN Forensics. Using the Adversarial Generative Network (GAN), many fake images or videos are completely generated instead of manipulated. This somehow reduces the performance of earlier detection approaches. Inspired by the camera fingerprints, recent researches try to analyze the fingerprints in generated images and explore the feasibility of attributing fake images to a GAN with certain architecture. Moreover, Zhang et al. proposed an AutoGAN to simulate artifacts produced by common GANs and detect GAN-generated images using spectrum features [10]. In [11], Cozzolino et al. attempted to spoof a smart pretrained embedder which is originally used to distinguish camera traces in capturing images. Their work revealed the vulnerabilities of current approaches. Durall et al. also investigated the artifacts left out of visual content. They analyzed the differences in the classical frequency domain and constructed 1D power spectrum statistics. Using this feature, a simple binary classifier trained with few annotated samples can achieve good performance [12]. Color abnormality is also a strong hint for GAN-generated content. In [13], McCloskey et al. demonstrated that GAN generators may leak some clues when converting feature representations to red, green, and blue pixels. Li et al. analyzed the difference between pristine and generated images in HSV, YCbCr, and RGB color spaces, and a statistical feature set was proposed to characterize the difference [14]. More directly, Nataraj et al. trained their CNN detector on cooccurrence matrices extracted from the RGB channels in the space domain and achieved competitive performance as well [15].

2.3. DeepFakes Forensics. Recently, many novel deep learning based technologies have also shown astonishing performance in face synthesis, among which the most famous is DeepFakes. Along with the continuous development of DeepFakes, the corresponding forensics technologies are also being researched. Similar to previous studies, early work mainly focused on detecting visual artifacts. Li et al. simulated the DeepFakes artifacts by Gaussian blur and affine warpage, and their evaluations indicated the simulated artifacts can make CNN detectors more robust [1]. Some other work focuses on dynamic defects in the temporal domain. In the pipeline of [3], a CNN is used as a spatial feature extraction backbone, and an RNN is connected to the backbone, aggregating the CNN outputs over time and makes final classifications. Zhou et al. aggregated short-term, long-term, and global statistics to characterize the relations among different face regions. Their evaluations indicated these relations, especially the temporal order within the tracklet, are informative for recognizing temporal inconsistency in manipulated face sequences [16]. Actually, most dynamic artifacts based detection approaches utilize a CNN backbone to firstly extract features of every single frame.

Facial expression habits are unique from person to person and are extremely hard to simulate. Therefore, DeepFakes may leave traces in respect to personality behavior habits and sometimes even the physical law of motions or illuminations. For example, by modeling the face and head movements as the unique speaking pattern of a specific individual, the high prediction error can be a strong hint of fake. Biological signals such as eye blinking and pulse are also discriminating cues to expose DeepFakes. Li et al. observed that the regular eye blinking cannot be realized in the synthesized videos, and they proposed a CNN and Long Short-Term Memory (LSTM) joint architecture to expose DeepFakes by predicting the eye blinking [17]. By the noncontact heart rate detection technology, it is easy to detect whether there is a regular heart rate in videos and identify the video authenticity. Similarly, Fernandes et al. proposed to estimate the heart rates in DeepFakes videos by Neural-ODE trained with normalized heart rate [18]. Due to insufficient datasets, the research on DeepFakes detection was seriously hindered in early time. To promote the research of DeepFakes detection, many large-scale datasets are made and open-sourced. Rossler et al. introduced a large facial manipulation dataset with 4k forged videos named Faceforensics++ created by four different approaches [19]. Recently, Facebook released a database containing 19154 pristine and 100K forged videos for the DeepFakes Detection Challenge (DFDC). There are various background conditions and manipulation approaches that are great challenges for detection approaches [20]. Li et al. proposed a new large scale benchmark named Celeb-DF that contains 5639 sophisticated DeepFakes videos [21]. Though some existing methods can expose fake videos, they generally make the video-level real/fake classification by fusing the predictions of several frames. This does not actually leverage the features of consecutive frames and leaves some room for fake video detection. To end this, we propose the CWSA module to accurately capture the temporal cues by fusing deep features of consecutive frames.

3. The Channel-Wise Spatiotemporal Aggregation

This section presents details of our proposed approach. Given a face patch sequence, the weights-sharing backbone extracts deep features of each patch. The proposed CWSA module then recombines the feature maps into a new feature sequence which is then compressed to a vector and connected to a single neural unit for real or fake classification. The complete pipeline of our approach is shown in Figure 1.

We propose a simple but effective module CWSA as in Figure 2. The proposed module is easy to cooperate with a backbone and serves as CWSA Net for face video forensics. Specifically, given a deep feature sequence of successive frames produced by the backbone, although it is unknown to us about the semantics of a specific channel, we hypothesis that feature maps of the same channels but different frames contain dynamic information as successive frames do. By stacking the feature maps of different frames with the same channel and carrying out further feature extraction separately, we can capture both frame-level artifacts and more refined interframe defects.

For every input frame, the backbone produces a feature map of size $F \in \mathbb{R}^{H\times W\times C}$, where H and W denote the resolutions and C denotes the channels. For a video clip that contains N successive frames, the weights-sharing backbone generates a set of feature maps of size $F' \in \mathbb{R}^{N\times H\times W\times C}$. Our approach firstly decomposes F' into base feature map $f_{n=1,c=1}^{N,C} \in \mathbb{R}^{H\times W}$ and recombines them by going through N and stacking f that has the equal c:

$$F_{new} = \left[f_{n=1,*}, f_{n=2,*}, \ldots, f_{n=N,*}\right]. \quad (1)$$

where $[\cdot]$ denotes channel-wise stacking. As in Figure 2, we finally get a new feature set with size $C \times F_{new} \in \mathbb{R}^{H\times W\times N}$, and in this paper, C equals to 1280 as we use EfficientNet B0 and H and W are both equal to 7.

The following up layers that deal with F_{new} are all weights-sharing, i.e., repeated C times to reduce the number of parameters. Batch Normalization is the first layer to avoid internal covariate shift that may seriously hindrance the training. Next, are convolution and LeakyReLU blocks with 128, 64, and 1 kernels with no downsampling and padding. A single feature map will happen to be converted to a single element, and regardless of the length of the input sequence, we will get a feature vector of size $i^{C\times 1}$. A single neural with sigmoid activation is connected to it and makes the classification fake or real. The pipeline of the proposed CWSA is summarized in Algorithm 1.

4. Evaluation and Discussion

4.1. Experimental Settings

4.1.1. Datasets and Preprocessing. In this work, we have carried out evaluations on a list of large scale fake face video

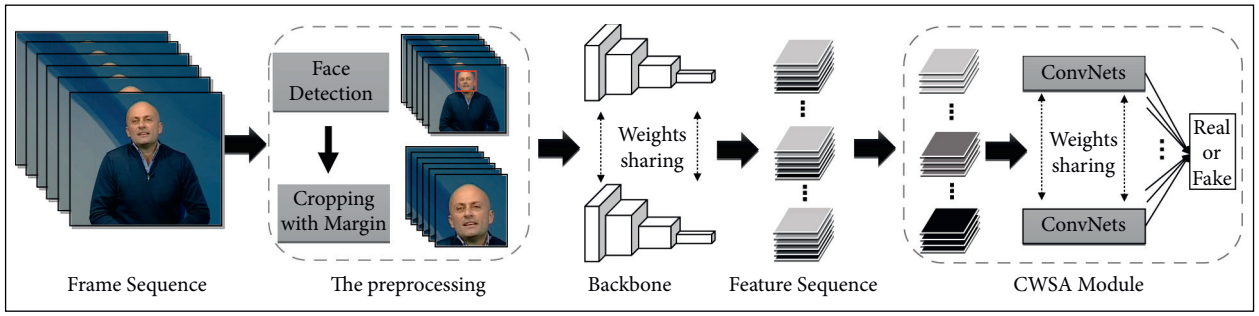

Figure 1: The proposed video forensics approach.

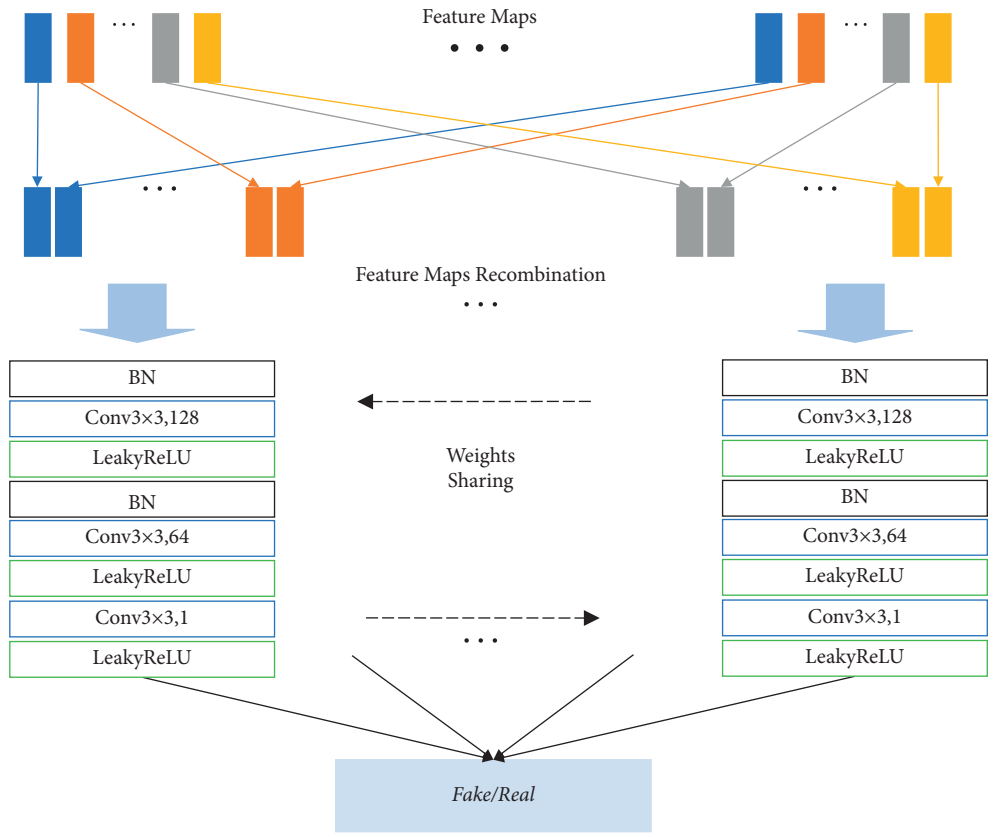

Figure 2: The architecture of the CWSA module; a colorful rectangle denotes a feature map output from the backbone.

Require: k Training face video clips $V_1, V_2, \ldots, V_k$; Corresponding label $y_1, y_2, \ldots, y_N$.
1: **for** each $i \in k$ **do**
2: Decompose V_i into the sequence of n frames $V_i^1, V_i^2, \ldots, V_i^n$;
3: Detect and crop faces frames from $V_i^1, V_i^2, \ldots, V_i^n$, then denote them as $V'^1_i, V'^2_i, \ldots, V'^n_i$;
4: **end for**
5: Feed $V'^1_i, V'^2_i, \ldots, V'^n_i$ into the backbone, producing a set of feature maps $F' \in \mathbb{R}^{N\times H\times W\times C}$;
6: Decompose F' into $f^{N,C}_{n=1,c=1} \in \mathbb{R}^{H\times W}$;
7: Combine f by going through N and stacking f that has the equal c, producing $C \times F_{new} \in \mathbb{R}^{H\times W\times N}$;
8: Feed F_{new} into weights-sharing classifier, producing y^{pred};
9: Calculating binary classification error between y and y^{pred};
10: Update the parameters of the model by back propagation;
Ensure: Optimal model for fake face detection

Algorithm 1: The CWSA algorithm.

datasets: FaceForensics++ [19], Celeb-DF [21], and DFDC Preview [20].

FaceForensics++ consists of 1000 pristine and 4000 forged videos evenly created by four different forgery methods: DeepFakes, Face2Face, FaceSwap, and NeuralTextures. In the following parts, we refer FaceForensics++ as FF++ and its subsets as DF, F2F, FS, and NT for simplicity.

Celeb-DF includes 590 pristine and 5639 forged videos created by advance DeepFakes technology. The source videos are publicly available YouTube video clips, including 59 celebrities of different genders, ages, and races. In this work, we use the second version of this dataset which contains another 300 pristine videos from YouTube.

Facebook DeepFake Detection Challenge Preview (DFDC-P) is the early release for this competition, which composes 1131 pristine of 66 actors, and 4113 forged videos created by two face synthesis algorithms.

Table 1 lists some more basic information about total frame numbers and video sizes.

Because the face region only makes up a tiny proportion in videos, it is necessary to crop off the face patches to reduce interference of redundant backgrounds and computation cost. Thus, we design a novel face cropping strategy which is proved to be beneficial for fake detection.

In the stage of preprocessing, we first detect faces in videos and then carry out face cropping, which raises a question about the optimal cropping strategy. In our earlier work, we naturally held that the characteristic of forged pixels is what a CNN mainly learns. In this case, we only have to crop off faces according to the results of face detection, and no more operations are required.

However, the evaluations reveal that by feeding inputs that include both pristine and forged pixels, deep CNNs can learn more about their difference. That is, networks may be benefited from this kind of input by detecting its global consistency. To validate if remaining some pristine pixels could help us with detection, we further evaluate two additional face cropping strategies for comparison:

(1) Crop off a convey hull of the detected face along with the key points of facial contour. The pixel density of other regions is set as 0.

(2) Crop off a minimal square that encloses the detected face with no extra margin.

(3) Crop off a minimal square that encloses the detected face and expand it by a factor of 1.4.

Samples of all three face cropping strategies are shown in Figure 3. Table 2 presents the performance between different face cropping strategies of image-level classification accuracy on EfficientNet B0. Obviously, by retaining more pristine pixels, the network is able to make some gain in accuracy. This is consistent on different datasets.

In order to verify the effect of the extended clipping factor on feature extraction, we add a simple ablation experiment based on EfficientNet B0. The experimental results on NT, which is a subset of FF++ and is a highly compressed version, are shown in Table 3. Obviously, the larger the margin is, the more the detection accuracy is, but the gain stops when expanding the original margin by a factor of 1.3. The detection accuracy decreases gradually with the increase of the factor, when it is greater than 1.3.

For this result, we think the reason is that the square of 1.3 is 1.69, whose half is about 0.85. When the square enclosing the detected face has no extra margin, the face region accounts for about 0.85 of the entire image region. Therefore, the factor of 1.3 makes the ratio between the number of face pixels and background pixels close to 1 : 1. That is, the ratio between the number of true and forged pixels is close to 1 : 1. We consider that preserving an appropriate proportion of true and forged pixels in the detection data is helpful to improve detection accuracy. We compare the accuracy convergence of different extended clipping factors in the training process of the whole model and display it in Figure 4. Weighing the accuracy and stability, we finally choose the factor of 1.4, which performs better and generalizes well to different datasets overall.

Specifically, given $F(x, y, w, h)$ that represent the coordinates of the upper left, width, and height of the detected face respectively α denotes the expanding coefficient that controls the size of the margin. We first compute the center position of this rectangle as shown in equation (2), and then generate new $F(x, y, w, h)$ as in equation (3), and accordingly cut off a square. We then resize the expanded faces to a uniform size regardless of the resolutions of the original videos, we set size = 224 and $\alpha = 1.4$ in this work.

$$P_c = \left(x + \frac{w}{2}, y + \frac{h}{2} \right), \tag{2}$$

$$\begin{aligned} \mathrm{Rect}_{\mathrm{new}} = F\Bigg(& P_c - \alpha \cdot \frac{\max(w,h)}{2}, \\ & P_c - \alpha \cdot \frac{\max(w,h)}{2}, \\ & \alpha \cdot \max(w,h), \\ & \alpha \cdot \max(w,h)). \end{aligned} \tag{3}$$

Considering that the head movements in videos are in a limited region in the short term, it is unwise to detect faces for every frame. Therefore, we only detect a portion of frames at regular intervals. For the undetected frames, we crop off faces by the detection results of the previously detected frame. In this paper, we detect faces for every 20 frames since the videos commonly contain 30 or 24 frames per second.

4.1.2. Hyperparameters. The performance is reported differently: frame-level accuracy is used to evaluate the performance of backbones that can only take single frames as input; video clip level accuracy is used to evaluate the models that take short sequences of consecutive frames. As in Table 4, the training of backbones and CWSA Net both consists of 40 epochs without early stopping. We use minibatch stochastic gradient descent as the optimizer and set *learning rate* = 0.01, *momentum* = 0.9, and learning rate decays by

Table 1: Basic information of datasets used in this work.

Dataset	Real/fake	Frames (k)	Size
FF++	1000/4000	509.9/2039.6	480p,720p,1080p
DFDC-P	1131/4113	88.4/1783.3	180p–2160p
Celeb-DF	890/5639	358.8/2116.8	Multiple

Figure 3: Results of three different face cropping strategies.

Table 2: Binary classification accuracy (%) of different face cropping strategies.

Cropping type	DF	NT	Celeb-DF
Convey hull	99.03	95.78	92.59
No-margin	99.09	97.34	91.88
1.4 × margin	**99.31**	**99.13**	**93.97**

2.375e-4 per epoch. For models trained on face images, *batch size* = 32 and *iterations* = 50. For CWSA Net, *batch size* = 16 due to memory limit and *iterations* = 100 for adequate training samples in each epoch. All performances are reported on 3200 random test samples.

In terms of evaluation metrics, we consider video forensics a binary classification task and adopt the metric binary classification accuracy that represents how many samples are correctly classified. Although pristine and forged videos in DFDC-P and Celeb-DF are unbalanced, we deliberately pick up samples of each class with a 50% probability to make it balanced in both training and testing. We also report the AUC (area under the curve) for comprehensive assessments.

Note that there is not any data augmentation used in this work. However, it is highly possible to achieve better results with appropriate augmentation, training hyperparameters, and other tricks. We choose not to do so because the aim of this work is to study the characteristics of deep models used for face forgery detection and the effectiveness of our approach.

4.2. Backbone Selection. The backbone is a key component that extracts deep features preliminarily. Thus, we systematically investigate the performance of different deep CNNs in fake face detection to determine the most task-orient one.

Table 3: Binary classification accuracy (%) of different face cropping strategies on highly compressed NT.

Cropping type	Compressed NT
1.1 × margin	75.32
1.2 × margin	75.64
1.3 × margin	**76.12**
1.4 × margin	75.48
1.5 × margin	74.30

EfficientNet B0 [5] shows its remarkable potential, and we consider it the backbone of our approach.

It is hard to design a face forensics task-oriented model from scratch. Although neural architecture search technology may help, it could lead to overfitting on specific datasets. The existing research on computer vision, especially general image classification on ImageNet, has provided some off-the-shelf deep models that perform preeminently on image feature extraction. However, their performance on ImageNet cannot be the only point of reference due to the difference between general image classification and forgery detection. It is not clear enough about how model architectures, internal modules, and layer combinations affect the detection performance. To this end, we systematically investigate the difference between various deep models.

As in Table 5, we evaluate a list of models and there is indeed some consistency when deep models are applied in forensics detection. Empirically, we chose models that can be divided by different standards. Considering *skip connections* and *inception modules* are the two most popular and effective components to construct modern deep models, the first standard classifies the chosen models by whether it contains *skip connections* (EfficientNet B0 [5] & Xception [22]) or not (Inception V3 [23] & MobileNet V1 [24]), and the second classifies by if the model contains *inception modules* (Inception V3 & Xception) or not (EfficientNet B0 & MobileNet V1). To classify real/forgery face patches, the output of the last convolution layer in all these models is compressed by a global average pooling to produce a feature vector, and a single neural unit with sigmoid activation is connected to it for classification.

It can be seen from Table 5 that *skip connection* is the key factor affecting detection accuracy. This is not evident enough on FF++ since the accuracies are saturated. On DFDC-P, it becomes more obvious, and the gap expands to 27% on Celeb-DF. A reasonable explanation is that low-level features help to expose facial manipulations, and *skip connection* can directly deliver these features to the downstream of models. To validate this, we remove the *skip connections* in EfficientNet B0 and Xception. On FF++, the performance of both models without *skip connections* seriously decreases and is even much worse than the other two. This is also can be seen on two other datasets, and their performance degrades to the same level as those models without *skip connections*. Overall, EfficientNet B0 performs best and generalizes well to different datasets and is an ideal backbone for image level feature extraction.

To further verify this, we define the variance distribution of the last dense layer as the *Neural Activity* of deep models:

$$\text{Neural Activity} = \left[\sum_{i=1}^{N}\frac{\left(O_i^1-\mu^1\right)^2}{N},\ldots,\sum_{i=1}^{N}\frac{\left(O_i^L-\mu^L\right)^2}{N}\right]^{\mathrm{T}}. \tag{4}$$

where for the dense layer with L units, O_i^l denotes the output of l-*th* neural unit of i-*th* sample, and μ^l denotes the mean output value of l-*th* neural over N samples. Because a neural unit of the last dense layer with a larger variance will contribute more to the final classification. Accordingly, an intensive *Neural Activity* indicates that most units are active at the same level and approximately equally contribute to final classification. We calculate the variances of every unit in the last dense layer of four models over $N = 3200$ test samples. Because performance on Celeb-DF is the most variable, thus we display the *Neural Activity* on four models by box plots in Figure 5. For Xception and EfficientNet B0, the first and the third quartiles are very close, which represents that their *Neural Activity* is very intensive, and most units contribute to detection. For the rest two, this range is relatively larger, which means there are many lazy units that contribute less to detection.

4.3. Performance of the Proposed Approach. On DFDC-P and Celeb-DF, we carry out sufficient experiments of different video clip lengths, and the results are shown in Tables 6 and 7, and the best results are shown in bold. The first column shows the performance of the EfficientNet B0 backbone on frames. Obviously, CWSA Net effectively improves the detection accuracy, and the longer the input sequence length is, the more the detection accuracy is. But this gain stops when the length is about 12 and is not continuing when the length further raises. As for parameters, the EfficientNet B0 backbone is the major part and contains about 4.05 M parameters. For the following layers, an input with 3 frames only needs extra 79234 parameters, and then for each additional frame, only additional 1152 parameters are required. We also evaluate the commonly used CNN-LSTM architecture with the same experimental hyperparameters. For the CNN-LSTM, it also leverages EfficientNet B0 as the backbone, and a 2048-unit LSTM takes the global average pooled outputs of the backbone which is a 1280-d vector. The LSTM is followed by a 512-d dense layer and a single neural to make the prediction. On DFDC-P, the CNN-LSTM even performs worse than the backbone. Although it indeed makes some gains on Celeb-DF that increases with sequence length, our CWSA Net still outperforms the CNN-LSTM.

In order to verify the superiority of the CWSA module, we compare it with a method that is a simple average fusion of each frame. In addition, we also compare the CWSA module with traditional RNN and LSTM based on the same feature extraction backbone (EfficientNet B0). Table 8 presents a comparison of the accuracy of these methods on NT, which is the subset of FF++. It should be noted that the NT used here is highly compressed, which is of lower quality. Obviously, the performance of RNN and LSTM is not very good, even not as effective as the simple average fusion of each frame. We consider that this is because RNN

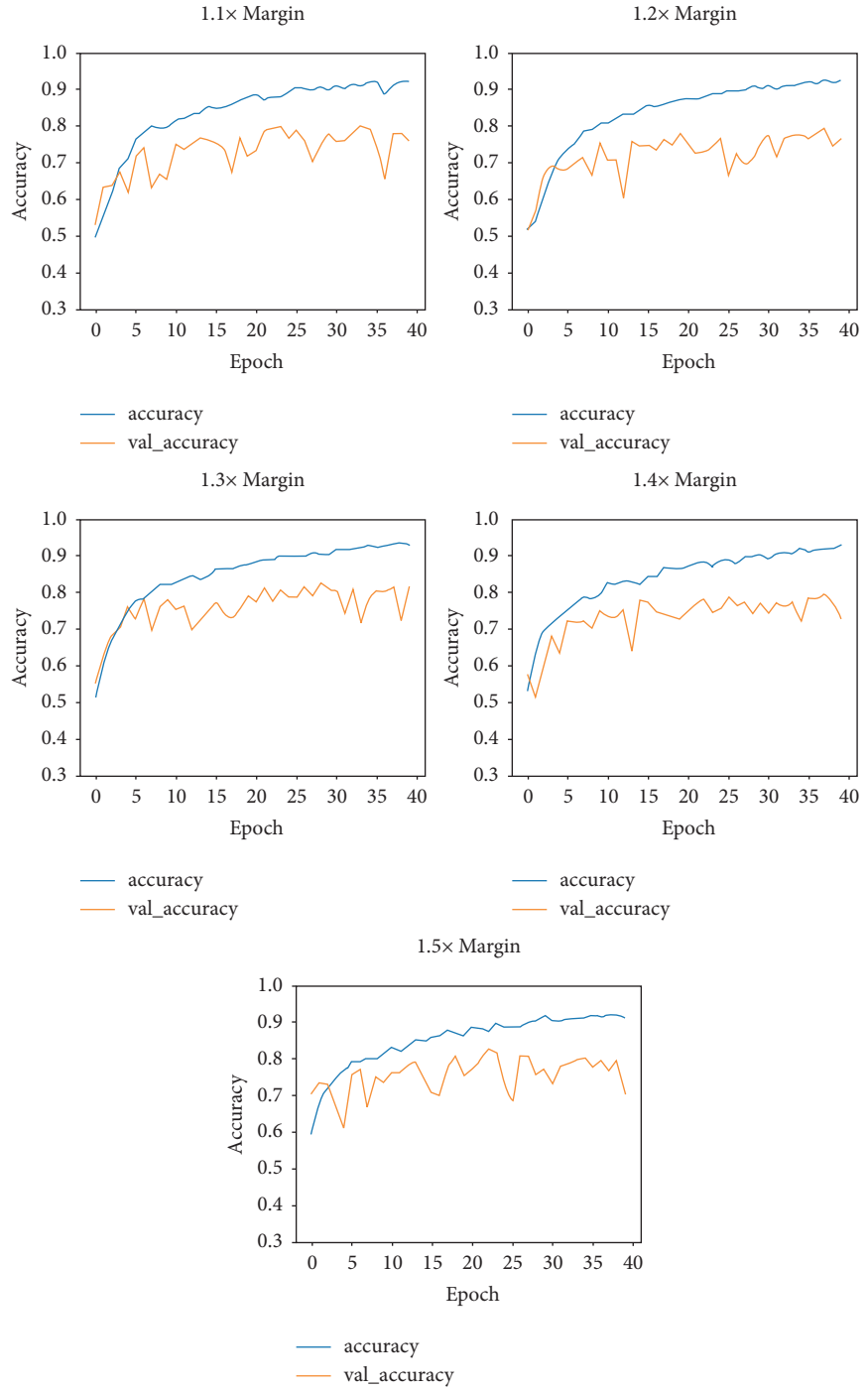

Figure 4: The comparison of accuracy convergence of different extended clipping factors.

Table 4: Training settings of different parts.

Settings	Training backbones	Training CWSA net
Batch size	32	16
Iteration	50	100
Epoch	40	40
Optimizer	SGD	SGD
Learning rate	0.01	0.01
Momentum	0.9	0.9
Decay	$2.375e-4$	$2.375e-4$

Table 5: Binary classification accuracy (%) (higher is better) of different backbones on frames.

Model	DF	F2F	FS	NT	DFDC-P	Celeb-DF
EfficientNet B0 [5]	**99.31**	99.69	99.53	**99.13**	**81.97**	93.97
Xception [22]	99.22	99.62	**99.56**	99.00	80.75	**94.84**
Inception V3 [23]	98.84	**99.78**	99.47	98.24	79.72	66.19
MobileNet V1 [24]	99.16	98.75	99.53	98.47	79.09	66.69
EfficientNet B0(w/o skip)	83.56	58.62	58.84	60.94	76.31	66.66
Xception (w/o skip)	94.91	58.80	64.62	53.91	65.44	67.50

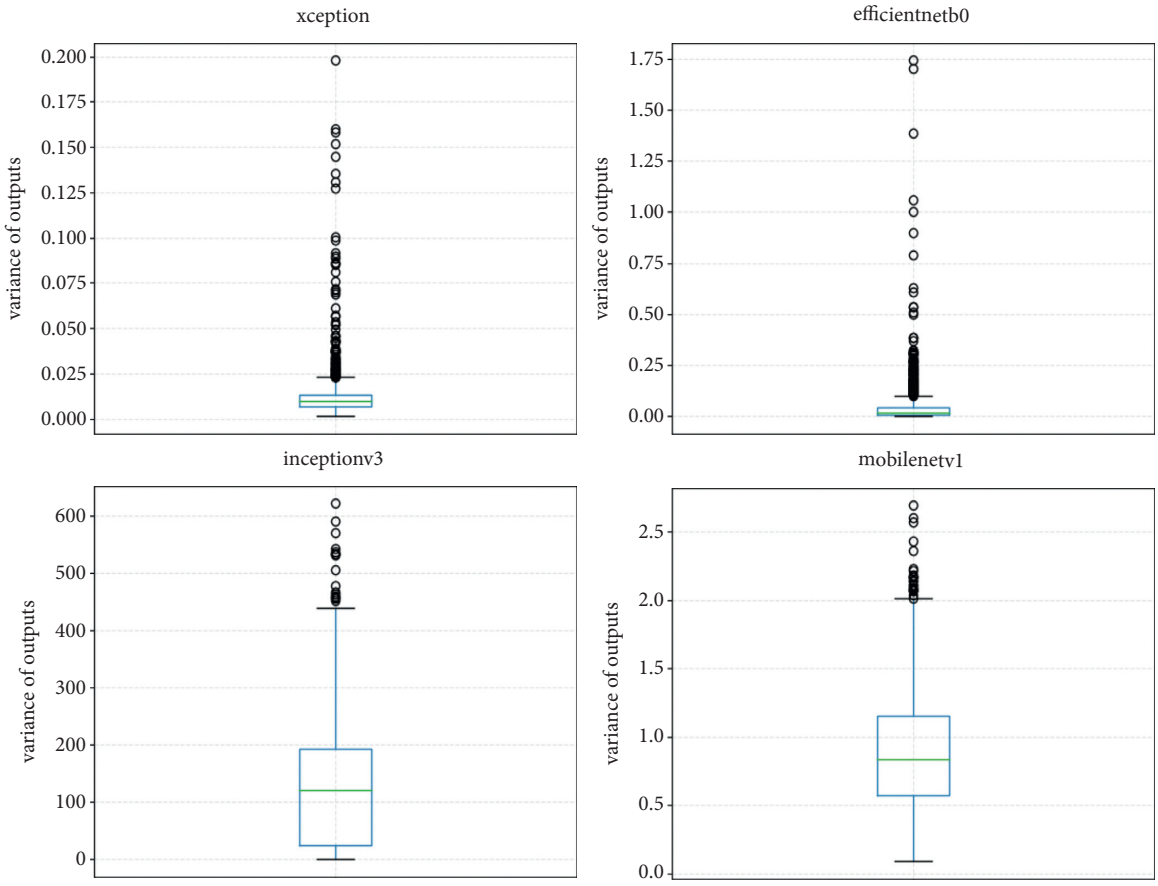

Figure 5: Neural activity of four models on Celeb-DF.

Table 6: Binary classification accuracy (%) (higher is better) on video clips on the DFDC-P dataset.

Model	1	3	6	9	12	15	18
CWSA net	81.97	84.76	83.14	82.75	**85.28**	84.81	83.19
CNN-LSTM	—	79.08	80.50	80.28	80.78	81.91	79.75

Table 7: Binary classification accuracy (%) (higher is better) on video clips on the Celeb-DF dataset.

Model	1	3	6	9	12	15	18
CWSA net	93.97	95.86	96.27	96.17	**97.12**	96.91	95.28
CNN-LSTM	—	95.22	95.06	95.13	96.53	96.38	95.28

and LSTM lose the spatial feature information within each frame when aggregating temporal features, resulting in a negative impact. From the experimental results, compared with the traditional RNN and LSTM, the proposed CWSA module performs better.

Table 8: Binary classification accuracy (%) (higher is better) comparison based on the same backbone on highly compressed NT datasets.

Approach	Compressed NT
CWSA	**80.6**
Simple fusion	79.4
LSTM	76.0
RNN	75.6

Table 9 presents a comparison of accuracy between our approach and the state-of-the-art on all three datasets. Although the CWSA Net is a little weaker on Celeb-DF, it is, on average, better than the state-of-the-art approach. Even on FF++, the accuracy almost saturates, CWSA Net still has a

Table 9: Binary classification accuracy (%) (higher is better) comparison of ours and the state-of-the-art approach on three datasets.

Approach	Celeb-DF	DFDC-P	FF++	Average
Ours	97.1	**85.3**	**99.4**	**93.9**
Biometric [25]	**98.5**	82.4	98.9	93.3

significant advantage. Also, it is worth noting that DFDC-P is obviously the most challenging dataset. Both methods are not very ideal in detection accuracy. However, CWSA Net still surpasses the state-of-the-art by 2.9%, which is a significant improvement.

A whole comparison based on AUC is in Table 10. There are various methods that derive from different perspectives on the list. CWSA Net achieves the highest AUC scores on both Celeb-DF and DFDC-P, demonstrating its efficiency. And obviously, compared to most of the other methods, CWSA Net improves the detection performance by a great gap.

We also compare the accuracy and AUC on all four subsets of FF++ with multiple detection methods in Tables 11 and 12. In this part, the results are provided by the EfficientNet B0 backbone only, and nothing expect the specially designed face cropping strategy is used. Apparently, our approach achieves the state-of-the-art level on average, and it goes beyond other methods on 3 out of 4 subsets. Although FF++ is rather easy to be exposed, and some of the previous methods perform nearly 100% in terms of both AUC and binary classification accuracy, our approach still shows obvious advantages on this dataset. These excellent results are not only because of the Efficient B0 but also because of the face cropping scheme presented in this work.

4.4. Analysis. We present some failure cases of forged face detection with our proposed approach on highly compressed NT, as shown in Figure 6. For the first type of failure case shown in Figures 6(a) and 6(b), it is obvious that the face detector fails to correctly extract the face from the highly compressed image, which directly degrades the detection accuracy of the forged face. Therefore, improving the robustness of the face detector can effectively solve such failures. For the second type of failure case, we adjust the color contrast of the images for a better display of the details and show them in Figures 6(c) and 6(d). Actually, color contrast is also one of the main factors affecting the detection of fake faces. In order to deal with such failures, digital image processing methods can be used to preprocess the samples with low color contrast. For the last type of failure case shown in Figures 6(e) and 6(f), the samples wrongly detected are video frames with different face poses. Due to the relatively few video frames of the side face in the datasets, the detection model is not sensitive to such samples, resulting in detection accuracy not being good enough. Therefore, the model can perform better with appropriate data augmentation.

Our work is one of the few existing methods in the field of fake face video detection that utilizes both airspace features and time domain features, and in Section 4.3, the experimental results have demonstrated its effectiveness. Differently, previous methods have focused on searching for clues of forgery at the image level [9, 19, 29]. Although these methods have had some success, they still leave room for improvement in terms of detection accuracy since they do not take advantage of significant temporal differences between real and fake as well. And our method attempts to further exploit temporal features to improve detection accuracy while maintaining the use of spatial features. In fact, we are not the first to consider this [3], but previous work destroyed the spatial structure of spatial features before extracting temporal features. This inevitably leads to a degradation of accuracy feature representation. Thus, the difference is that the proposed method extracts spatial-temporal features at the same stage, which mitigates the deterioration of spatial features, leading to advantages across multiple datasets.

Table 10: AUC (higher is better) comparison on Celeb-DF and DFDC-P datasets.

Approach	Celeb-DF	DFDC-P
Ours	**0.997**	**0.925**
Metric learning [26]	0.992	—
Face X-ray [27]	0.748	—
Fakespotter [28]	0.668	—
Mesonet [29]	—	0.753

Table 11: Binary classification accuracy (%) (higher is better) comparison on FF++ datasets.

Approach	DF	F2F	FS	NT	Average
Ours	**99.31**	**99.69**	**99.53**	**99.13**	**99.42**
Xception [19]	—	—	—	—	99.26
Fakespotter [28]	—	—	—	—	98.50
CNN-RNN [3]	96.90	94.35	96.30	—	95.85

Table 12: AUC (higher is better) comparison on FF++ datasets.

Approach	DF	F2F	FS	NT	Average
Ours	**0.997**	**1.0**	**0.999**	0.988	**0.996**
CNN-RNN [3]	0.996	0.984	0.994	—	0.991
Face X-ray [28]	0.992	0.991	0.992	**0.989**	0.991
Camera noise [9]	0.963	0.939	0.978	—	0.960

4.5. Industrial Applications. Currently, the negative effects of the fake face videos mainly remain on the network, as presented in Figure 7, and due to the constraints of laws and policies, they are not too excessive to bring serious adverse effects. However, these face manipulation technologies are nonnegligible threats to the systems that rely on face recognition in real word, not just in cyber world. A normal face recognition system without a strong face antispoofing module often requires the user to make corresponding facial movements as instructed to verify his legitimacy. If this step is passed, the system will retrieve the captured faces from a local or cloud-based database to further determine whether he/she is authorized or not. But this kind of face recognition system without forgery algorithms or modules usually cannot resist face-swap attacks.

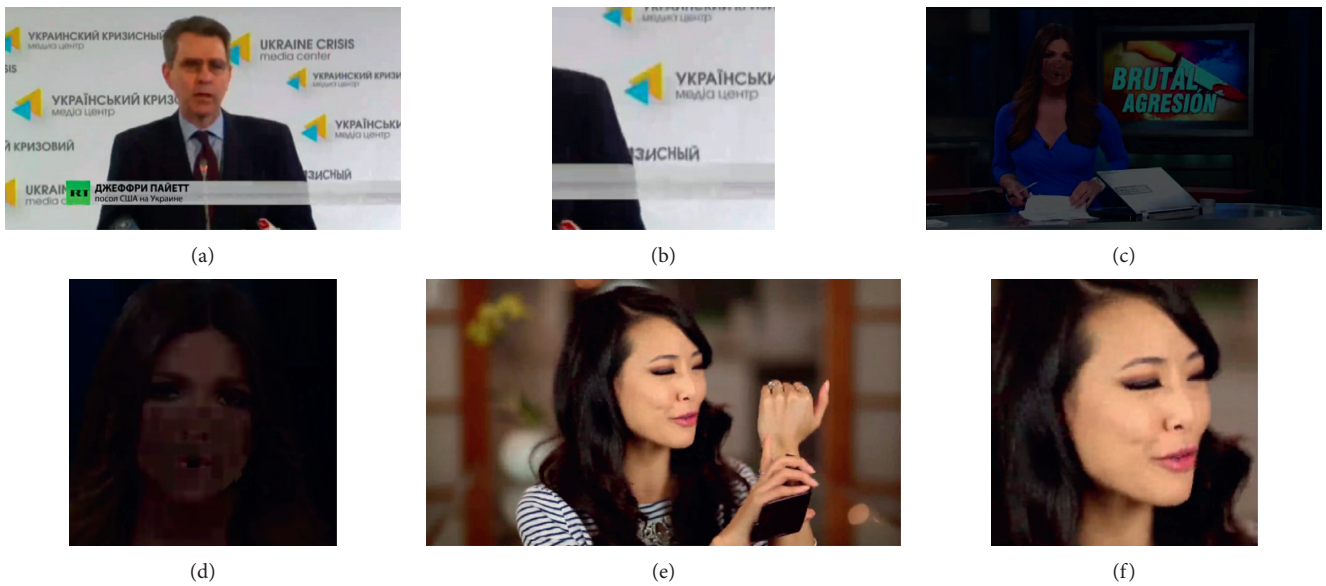

Figure 6: Failure cases on highly compressed NT. (a, c, e) Video frame. (b, d, f) Extracted face.

Figure 7: Fake faces videos circulating on the Internet [30].

Face recognition technology, due to its convenience and remarkable, has been applied in a few interactive intelligent applications. In those scenarios with high security requirements, these easily exposed face recognition systems have a number of security implications. Exiting face recognition systems are vulnerable to presentation attacks ranging from makeup, print, 3D-mask, etc. In recent years, in order to ensure the security of face recognition systems, face antispoofing (FAS) technology is also highly concerned [31]. Yu et al. proposed the first FAS method based on neural architecture search to discover the well-suited task-aware networks [32]. However, the forged face can also indirectly attack the face recognition system in these ways, which can hardly be ignored. The hacker may leverage the face-swap algorithms to simulate the facial movements following the instruction and print or display the forged face on some medium like paper or electronic screen in order to deceive the system. This calls the requirement of an additional fake face detection module in the first phase of the face recognition system to eliminate the safety hazards, as shown in Figure 8. More importantly, the forgery algorithm in the real-life scenario is unknown, and the detection algorithm needs to be highly robust to multiple forgery types. The CSWA tested on benchmark containing different types of face swapping and reenactment, which are both capable of assaulting face recognition systems, can assist these systems in defending these attacks. To ensure user experience, face recognition systems usually require the entire pipeline to be relatively fast, so the face forgery detection module can only acquire a short video of the face, but our method only requires a limited number of frames to achieve high precision detection.

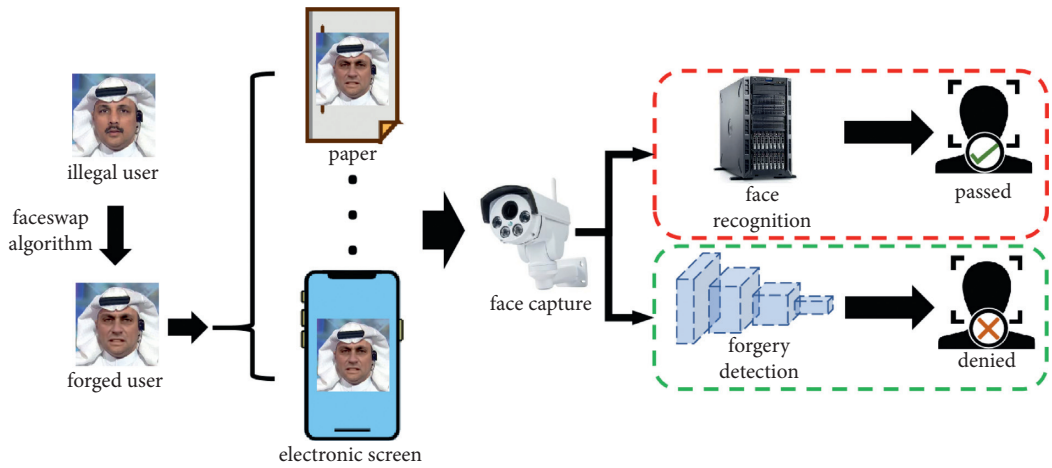

Figure 8: Detecting fake face attack with the forgery detection module.

5. Conclusion and Future Work

In this paper, we describe a novel forensic module named CWSA to detect face video manipulations. To take a close look at the problem of manipulation detection using deep CNNs, we first study the influence of face cropping strategies and architectures of different networks. We find that in face cropping, a suitable margin helps models perform better. And *skip connections* that pass low-level features downstream are also very beneficial in this task. On these bases, we propose our simple but smart CWSA Net that recombines feature maps belonging to the same channel from consecutive frames and fuses them by separately convoluting the new feature map sets. Our approach is demonstrated to be very competitive by the evaluations on three large-scale face video manipulation benchmarks. It achieves the state-of-the-art level on average and goes beyond other methods on most of the datasets. On the most challenging dataset DFDC-P, the performance of both our and the state-of-the-art approaches is not very ideal but the CWSA Net still surpasses it by 2.9%, which is a significant improvement.

Our work indicates some opportunities for future research, as it proves the feasibility of detecting forged faces from spatial and temporal perspectives. Firstly, although the CWSA module aggregates interframe features without destructing their spatial structure, there is no further constraint on the interesting regions. That is, the module treats different locations of features equally. Intuitively, the amount of information about forgery flaws exposed in the time domain varies across regions, and the more informed regions are supposed to be more focused. Thus, we can turn to the attention mechanism, but due to the lack of ground truth of interested regions, we have to design an attention module in an unsupervised or semisupervised manner. Another opportunity for future work is domain generalization. Existing detection approaches, including ours, are not robust enough to unknown types of fakes, but facing unknown attacks is common in real-life scenarios, and the ability to generalize to an unknown domain is essential if we want our approach to be more pragmatic. To this end, in future work, we expect our approach to be more than just a binary classifier, but to aggregate real faces through metric learning while making the fake face as separate from the real face as possible, in this case, the fake face detection tasks is viewed as anomaly detection tasks.

Data Availability

The data that support the findings of this study are openly available in Yujiang-Lu/CWSA-tensorflow at https://github.com/Yujiang-Lu/CWSA-tensorflow.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported in part by the Jiangsu Basic Research Programs-Natural Science Foundation under grant no. BK20181407, in part by the National Natural Science Foundation of China under grant nos. U1936118, 61672294, U1836208, 61702276, 61772283, 61602253, and 61601236, in part by Six Peak Talent Project of Jiangsu Province (R2016L13), Qinglan Project of Jiangsu Province, and "333" Project of Jiangsu Province, in part by National Key R&D Program of China under grant 2018YFB1003205, in part by the Priority Academic Program Development of Jiangsu Higher Education Institutions (PAPD) fund, and in part by the Collaborative Innovation Center of Atmospheric Environment and Equipment Technology (CICAEET) fund, China.

References

[1] Y. Li and S. Lyu, "Exposing deepfake videos by detecting face warping artifacts," in *Proceedings Of the IEEE Conference On Computer Vision And Pattern Recognition Workshops*, pp. 46–52, Long Beach, California, 2019.

[2] E. Zakharov, A. Shysheya, E. Burkov, and V. Lempitsky, "Few-shot adversarial learning of realistic neural talking head models," in *Proceedings Of the IEEE International*

Conference On Computer Vision, pp. 9459–9468, Seoul, korea, October 2019.

[3] E. Sabir, J. Cheng, A. Jaiswal, W. AbdAlmageed, I. Masi, and P. Natarajan, "Recurrent convolutional strategies for face manipulation detection in videos," *Interfaces*, vol. 3, p. 1, 2019, https://arxiv.org/abs/1905.00582.

[4] I. Amerini, L. Galteri, R. Caldelli, and A. Del Bimbo, "Deepfake video detection through optical flow based cnn," in *Proceedings Of the IEEE International Conference On Computer Vision Workshops*, Seoul, Korea (South), October 2019.

[5] M. Tan and Q. Le, "Efficientnet: rethinking model scaling for convolutional neural networks," in *Proceedings International Conference On Machine Learning*, pp. 6105–6114, PMLR, Long Beach, California, June 2019.

[6] D. Cozzolino, G. Poggi, and L. Verdoliva, "Splicebuster: A new blind image splicing detector," in *Proceedings of IEEE International Workshop on Information Forensics and Security (WIFS)*, pp. 1–6, IEEE, Rome, Italy, November 2015.

[7] S. Mandelli, N. Bonettini, P. Bestagini, V. Lipari, and S. Tubaro, "Multiple jpeg compression detection through task-driven non-negative matrix factorization," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 2106–2110, IEEE, Calgary, Alberta, Canada, April 2018.

[8] J. Luka, J. Fridrich, and M. Goljan, "Digital camera identification from sensor pattern noise," *IEEE Transactions on Information Forensics and Security*, vol. 1, no. 2, pp. 205–214, 2006.

[9] D. Cozzolino, "Extracting camera-based fingerprints for video forensics," in *Proceedings Of the IEEE Conference On Computer Vision And Pattern Recognition Workshops*, pp. 130–137, Long Beach, CA, USA, June 2019.

[10] X. Zhang, S. Karaman, and S.-F. Chang, "Detecting and simulating artifacts in gan fake images,," in *Proceedings of IEEE International Workshop on Information Forensics and Security (WIFS)*, pp. 1–6, IEEE, Hong Kong, China, December 2019.

[11] D. Cozzolino, J. Thies, A. Rössler, M. Niezner, and L. Verdoliva, "Spoc: spoofing camera fingerprints," 2019, https://arxiv.org/abs/1911.12069.

[12] R. Durall, M. Keuper, F.-J. Pfreundt, and J. Keuper, "Unmasking deepfakes with simple features," 2019, https://arxiv.org/abs/1911.00686.

[13] S. McCloskey and M. Albright, "Detecting gan-generated Imagery using color cues," 2018, https://arxiv.org/abs/1812.08247.

[14] H. Li, B. Li, S. Tan, and J. Huang, "Detection of deep network generated images using disparities in color components," 2018, https://arxiv.org/abs/1808.07276.

[15] L. Nataraj, T. M. Mohammed, B. S. Manjunath et al., "Detecting gan generated fake images using cooccurrence matrices," *Electronic Imaging*, vol. 2019, no. 5, pp. 532–1–532–7, 2019.

[16] T. Zhou, W. Wang, Z. Liang, and J. Shen, "Face forensics in the wild," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5778–5788, New Orleans, US, November 2021.

[17] Y. Li, M.-C. Chang, and S. Lyu, "In Ictu Oculi: Exposing Ai Created Fake Videos by Detecting Eye Blinking," in *Proceedings of 2018 IEEE International Workshop on Information Forensics and Security (WIFS)*, pp. 1–7, Hong Kong, June 2018.

[18] S. Fernandes, S. Raj, E. Ortiz et al., "Predicting heart rate variations of deepfake videos using neural ode," in *Proceedings of the IEEE International Conference on Computer Vision Workshops*, Seoul, South Korea, October 2019.

[19] A. Rossler, D. Cozzolino, L. Verdoliva, C. Riess, J. Thies, and M. Nie, "Faceforensics++: learning to detect manipulated facial images," in *Proceedings Of the IEEE International Conference on Computer Vision*, pp. 1–11, Seoul, South Korea, November 2019.

[20] B. Dolhansky, R. Howes, B. Pflaum, N. Baram, and C. C. Ferrer, "The deepfake detection challenge (dfdc) preview dataset," 2019, https://arxiv.org/abs/1910.08854.

[21] Y. Li, X. Yang, P. Sun, H. Qi, and S. Lyu, "Celeb-df: a new dataset for deepfake forensics," 2019, https://arxiv.org/abs/1909.12962.

[22] F. Chollet, "Xception: deep learning with depthwise separable convolutions," in *Proceedings of 2017 IEEE Conference on Computer Vision and Pattern Recognition*, July 2017.

[23] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proceedings Of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2818–2826, Las Vegas, NV, USA, June 2016.

[24] A. G. Howard, M. Zhu, B. Chen et al., "Mobilenets: efficient convolutional neural networks for mobile vision applications," 2017, https://arxiv.org/abs/1704.04861.

[25] S. Agarwal, T. El-Gaaly, and H. Farid, "Detecting deep-fake videos from appearance and behavior," 2020, https://arxiv.org/abs/2004.14491.

[26] A. Kumar, A. Bhavsar, and R. Verma, "Detecting deepfakes with metric learning," in *2020 8th International Workshop On Biometrics And Forensics (IWBF)*, pp. 1–6, IEEE, Porto, Portugal, April 2020.

[27] L. Li, J. Bao, T. Zhang et al., "Face x-ray for more general face forgery detection," in *Proceedings Of the IEEE/CVF Conference On Computer Vision And Pattern Recognition*, pp. 5001–5010, Seattle, WA, USA, June 2020.

[28] R. Wang, F. Juefei-Xu, L. Ma et al., "Fakespotter: A simple yet robust baseline for spotting ai-synthesized fake faces," in *Proceedings of International Joint Conference On Artificial Intelligence (IJCAI)*, Yokohama, Japan, 2020.

[29] D. Afchar, V. Nozick, J. Yamagishi, and I. Echizen, "Mesonet: a compact facial video forgery detection network," in *Proceedings of IEEE International Workshop on Information Forensics and Security (WIFS)*, pp. 1–7, IEEE, Hong Kong, March 2018.

[30] Fake faces videos circulating on the internet. [Online]. Available: https://www.theguardian.com/technology/2020/jan/13/what-are-deepfakes-and-how-can-you-spot-them.

[31] Z. Yu, Y. Qin, X. Li, C. Zhao, Z. Lei, and G. Zhao, "Deep learning for face anti-spoofing: a survey," 2021.

[32] Z. Yu, J. Wan, Y. Qin, X. Li, and G. Zhao, "Nas-fas: static-dynamic central difference network search for face anti-spoofing," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, no. 9, pp. 3005–3023, 2020.