Cascaded-Hop For DeepFake Videos Detection

Dengyong Zhang[1,2], Pengjie Wu[1,2], Feng Li[1,2]*, Wenjie Zhu[1,2], and Victor S. Sheng[3]
[1] Hunan Provincial Key Laboratory of Intelligent Processing of Big Data on Transportation, Changsha University of Science and Technology, Changsha, 410114, China
[Email: zhdy@csust.edu.cn, lif@csust.edu.cn]
[2] School of Computer and Communication Engineering, Changsha University of Science and Technology, Changsha, 410114, China
[3] Department of Computer Science, Texas Tech University, Lubbock, 79409, TX, USA
*Corresponding author: Feng Li

Received January 26, 2022; revised April 26, 2022; accepted May 12, 2022; published May 31, 2022

Abstract

Face manipulation tools represented by Deepfake have threatened the security of people's biological identity information. Particularly, manipulation tools with deep learning technology have brought great challenges to Deepfake detection. There are many solutions for Deepfake detection based on traditional machine learning and advanced deep learning. However, those solutions of detectors almost have problems of poor performance when evaluated on different quality datasets. In this paper, for the sake of making high-quality Deepfake datasets, we provide a preprocessing method based on the image pixel matrix feature to eliminate similar images and the residual channel attention network (RCAN) to resize the scale of images. Significantly, we also describe a Deepfake detector named Cascaded-Hop which is based on the PixelHop++ system and the successive subspace learning (SSL) model. By feeding the preprocessed datasets, Cascaded-Hop achieves a good classification result on different manipulation types and multiple quality datasets. According to the experiment on FaceForensics++ and Celeb-DF, the AUC (area under curve) results of our proposed methods are comparable to the state-of-the-art models.

Keywords: Face manipulation, Deepfake detection, Machine learning, PixelHop++, SSL

This work was supported in part by National Natural Science Foundation of China (62172059, 62072055), Hunan Provincial Natural Science Foundations of China (2020JJ4626), Scientific Research Fund of Hunan Provincial Education Department of China under Grant (19B004), Postgraduate Scientific Research Innovation Project of Changsha University of Science and Technology (CX2021SS76).

http://doi.org/10.3837/tiis.2022.05.014 **ISSN : 1976-7277**

1. Introduction

As the most influential technique in applying the deep learning to face manipulation, Deepfake, shown in **Fig. 1** and **Fig. 2**, had become a synonym for face manipulation techniques [1]. Deepfake has attracted so much attention in recent years because it has caused a series of social security issues that the government, enterprises, and even ordinary citizens cannot ignore.

On one hand, the face is the most widely used to bind between smart devices and its user's biological identity information [2] in many scenarios (unlock the smartphone, ID verified in bank or custom). Manipulating someone's face means stealing her or his biological identity, which may cause people's loss of property or reputations. On the other hand, face manipulation is closely related to internet security. If a large number of videos and images containing fake face information (especially faces of influential people, such as government officials, and the business elite) are widely spread on the internet may bring great problems such as fake news, fraud and financial fraud [3, 4].

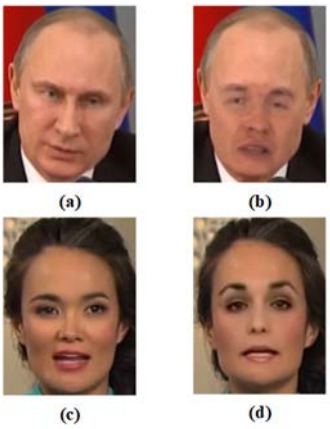


Fig. 1. Frames from FaceForensics++. (a), (c) are real. (b), (d) are fake

Fig. 2. Frames from Celeb-DF (v2) and generated by the Deepfake method.

As a result, the field of forensics research is being encouraged to Deepfake detection in images and videos [5-9]. It is no doubt that face manipulation detection solutions using deep learning (DL) has made great achievements. For instance, models [10-14] are based on convolutional neural networks (CNN), and models [15-17] are based on integrated CNN and recurrent neural networks (RNN). In addition, generative adversarial networks [18, 19] (GAN) have already become an increasingly significant player in this field. Those models [20-24] are always able to achieve high detection accuracy. Lastly, there are also many non-DL-based models [25-27], but they generally require handcrafted datasets as the source of classification features. The most salient question about those detection methods is that their effectiveness will be limited to the manipulation methods [12, 27-28].

We describe a method for preprocessing video datasets containing faces and propose a Deepfake detector based on non-DL. Some previous works [11, 14, 26] have proved that whether a model is DL-based or non-DL-based, how preprocessing the dataset is a very important step. Different from other methods of extracting facial frames from videos, we are not saving one frame every five, six, or other static number frames, but iterate through all frames of a video and save proper size frames with clear faces, and faces are in a certain

distance. The other point of this preprocessing method is that rather than directly resize frame images like other methods, we use the residual channel attention network (RCAN) [29] to achieve the goal of restoring high-frequency features and adjusting images size. Through such preprocessing, the datasets of Deepfake sample frames we made can not only retain as much facial information as possible from a video but also avoid the redundant influence of too many repeated facial images when training the Deepfake detector. The model we proposed for the Deepfake videos detection solution is called Cascaded-Hop. Cascaded-Hop is based on PixelHop++ system [30] and successive subspace learning (SSL) principle [31-33]. Following the idea of SSL, we use the PixelHop++ module to extract the most important features from multiple regions of high-resolution color facial images for classification. Then judge the authenticity of a video by ensemble pre-classifier labels.

Simply speaking, this paper has two main contributions: A preprocessing method that can get clearer and low repeatability Deepfake sample frames; A Deepfake detector based on SSL which can classify Deepfake videos effectively on multiple public datasets.The rest of the manuscript is organized as follows: Background review of face manipulation and detection techniques in Sec. 2. We describe our contributions in detail including the images/videos preprocessing of dataset and the Cascaded-Hop model in Sec. 3. Particularly experiments on multiple Deepfake video datasets in Sec. 4. Lastly, we have a brief conclusion in Sec. 5.

2. Background Review

2.1. Deepfake Techniques

The Deepfake algorithm became widely known in 2017 by a Reddit user named Deepfakes, and it is usually used to change one's face in film and television works. Different from traditional facial manipulation, this term has attracted amazing social attention since its birth [34, 35] because it had led to a revolutionary development in face manipulation. Since then, Deepfake techniques, especially those that used deep learning, have boosted development. The number of fake images and videos that are hard to distinguish had spread on social media platforms was too much to count. At the same time, it also threatens the information security of intelligent device users.

The manipulation can be defined into four types [1]: face synthesis, face swap, facial attributes tempering, and facial expression replay. Face synthesis means the hacker change one person's facial information with another face which may be non-existent in the real world and they are usually created by the GAN-based method [36]. Face swap means replacing a face in an image or video with another people's face. There are three most famous techniques used in this facial temper: Deepfake, FaceSwap, and ZAO. The third type of manipulation generally aims to modify attributes (the hair and skin color, age, or gender) of a face. GAN-based models are also used for facial attributes manipulation like StarGAN [37] and RecycleGAN [38]. The last type of Deepfake manipulation method is applied to change one's facial expression by transferring another people's expression. The most popular technique is called Face2Face [39].

Even though these Deepfake techniques have a high forgery level, they always leave identifiable trace features different from real media for classification. For example, in [25], the authors argue that Deepfake algorithms can usually only generate fixed-size facial images due to the limited computational resources and production time. To improve the quality of the fake videos, the generated source video must undergo affine distortion to better merge it into the target video.

2.2. Deepfake Detectors

In the light of the principle and idea of different detectors, the Deepfake detectors can be classified into three types: non-DL-based, CNN-based, and GAN-based.

2.2.1 Non-DL-based methods

The main idea of those non-DL-based facial tempering detection models is that the manipulation process can be regarded as the generators blending source faces into existing target facial images or videos. Face X-ray [27] is a representative detector of non-DL methods and this model got a nice classification result in FaceForensics++. The authors had described that blend processing will bring their unique traces introduced from the hardware level (such as sensor models and spectral characteristics) or software level (such as compression or composition algorithms) into target images or videos. 3D-based models are also used in face manipulation detection like [25, 40]. Especially in the work of [25], the researchers had proposed a 3D-based head pose estimation model and experiments on the UADFV dataset and DARPA dataset which demonstrated that discrepancy between head pose and facial landmarks can be revealed when the 3D head poses are well estimated in manipulated facial images. In the work of [41], the detector also uses a PixelHop++ model based on SSL, which is evaluated on the UADFV and Celeb-DF to get a better Deepfake classification result.

2.2.2. CNN-based methods

The DL-based detector is the most widely used for face manipulation detection due to the deep learning technology being relatively mature, computer hardware resources (CPU, GPU) supporting the CNN framework (ResNet50, ResNet101, VGG16, and so on) are easy to get [42-44], and many open-source datasets can be used for training and testing models. Tolosana *et al*. [10] using an Xception network certify that different regions and landmarks in the face can as the units for Deepfake classification. Peng Zhou *et al.* [11] contributed a two-stream architecture that can learn both high-level pampering artifacts and low-level noise features by incorporating GoogLeNet and triplet networks. Afchar *et al*. [12] contributed a mesoscopic method and created models which are based on Meso-4 and MesoInception-4 networks for manipulated videos. Huy H. Nguyen *et al*. [13] have designed a Y-shaped decoder which is a multi-task learning model structure, and this detector can be used to classify and segment facial information simultaneously. The method in [14] is proposed to detect the warping trace appearing when a face is warped. Many other works [45-47] also achieved a nice effect in this field.

Compared with non-DL methods, one of the advantages of using DL is that the temporal and spatial features of videos can be well utilized. For example, the works of [48, 49] use the people's eye blink continuity of the frame sequences of a video as the main features for Deepfake videos detection. Moreover, some RNN-based models like [15] make full use of the temporal correlation of video streams. An optimal strategy was proposed in [15] to combine the temporal information in the video stream with face preprocessing technology. In the work of [26], Deepfake videos are classified by extracting classification features from the eyes, teeth, and facial contours.

2.2.3. GAN-based methods

The ideology of GAN had made an epoch-making impact in making something out of nothing. Generators who are based on GAN can volume create very realistic facial images. The work [20] exploited the color spaces disparity between GAN-generated images and real ones. S.

McCloskey and M. Albright's work [22] via experiments showed that the color processing of the GAN-based generators is significantly different from the real camera in two aspects. The work of J.C. Neves *et al.* [23] proved that the images generated by GAN will leave inherent fingerprint features of the generators and proposed a model to remove fingerprint features to evaluate many Deepfake classification models. Though CNN-based and GAN-based Deepfake detectors have achieved very good classification results in many public dataset benchmarks, they are data-driven architectures that mean all those detection models need to be trained and tested by feeding a large number of data.

Compared with the previous work, our work is carefully select Deepfake sample frames in the preprocessing stage, and the direct result is to get a higher visual quality Deepfake sample dataset. In addition, our Deepfake detector extract features from the sub-regions of images and steps up to video-level classification which had different motivations from many previous studies.

3. Proposed Method

3.1. Face Images/Videos Preprocessing

Public large-scale datasets of Deepfake videos such as the well-known FaceForensics++ (FF++) [8] are massively generated by automatic generators. So, there are many videos with low-quality frames. If we watch them by playing a certain number of videos in FF++, it would be found that the videos contain the fuzzy, repeat, face twisted, covering incomplete, which we defined as low-quality frames.

Unlike many preprocessing methods that simply and blindly save one frame every few frames from a video. We proposed a method that saves facial frames from videos through a series of rigorous processes. To get higher quality Deepfake samples while eliminating some low-quality frames of a video, our work never discards a frame unreasonably or randomly.

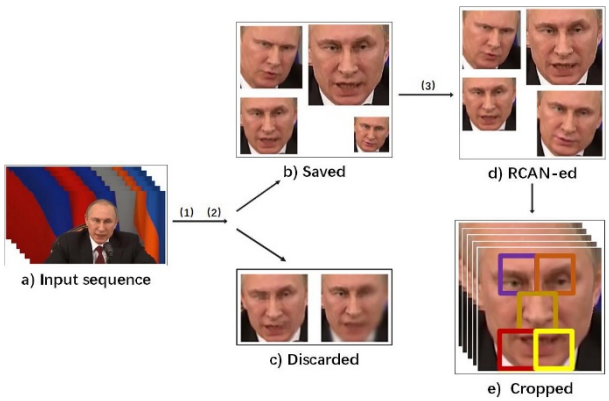


Fig. 1. Diagram of our preprocessing method. (1) and (2) are the algorithms to discard low-quality frames. (3) is the method of RCAN.

Fig. 2. Result of our preprocessing method. Fames like the first row will be saved and frames like the second row will be discarded.

Firstly, as shown in **Fig. 3**, our method will traverse the whole frame sequence of a video to find the frames with facial images and eliminate ones whose facial size is smaller than 68×102. The reason for this preprocessing is those smaller frame images always bring little features and are unsuitable to participate in our subsequent work. The saved and discarded frame images are exampled in **Fig. 4**.

Secondly, according to the basic geometric meaning of image matrix, we defined the space vector distance of adjacent frames as $\|Đ\|_2$, see (1):

$$\|Đ\|_2 = \sqrt{\sum_{i=1}^{n}\left(V_{A_i} - V_{B_i}\right)^2} \tag{1}$$

Where V_{A_i} is the i-th pixel value of the former facial gray image's matrix and the V_{B_i} is the latter one's i-th pixel value. While the result value of $\|Đ\|_2$ is smaller than 0.3 which means that the space vector distance between A and B is too close to be seen as repetitive.

Whenever meeting two repetitive frames and need to save only one of them, we compare their sharpness using the Laplacian algorithm which is defined as Eq. (2):

$$Img_{clt} = \begin{cases} A, \ Lap_A.Var \geq Lap_B.Var \\ B, \ Lap_A.Var < Lap_B.Var \end{cases} \tag{2}$$

Where Img_{clt} is the clearer facial image between image A and image B, $Lap_A.Var$ and $Lap_B.Var$. are the values of the sharpness measurement index of A and B. Those values were obtained from the Laplacian algorithm which can be used to analyze the sharpness of images. By doing so, we can save the clearer frames from multiple similar images.

Through the previous two processes, we would save clearer and nearly without repeating facial images from the video. In this way, we eliminated the low-quality facial images from many Deepfake datasets successfully and made datasets redundancy avoided to a great extent.

As a result, the number of frames selected from a video can be controlled in about 20 and 40. To satisfy the subsequent requirements of the Cascaded-Hop based on SSL, instead of directly resizing the small facial images, we use an image super-resolution method named RCAN to enlarge images. RCAN can restore the high-frequency and low-frequency information of an image by adjusting features adaptively through the interdependence of feature channels. In this way, RCAN can not only expand the size of the smaller images but also yield features for Cascaded-Hop to learning.

The last step of preprocessing is to crop face images and save multiple regions. In many works [24, 51], it has been proved that the rich features of the tampered images will be particularly prominent in the eyes, nose, and mouth. So, we select these areas as the cropping regions according to the facial landmark. The effect is shown roughly in the image (e) of **Fig. 3,** where the five different colored boxes will be covering the left eye, right eye, nose, and mouth, but may not completely cover them due to the different sizes of the images.

3.2. Feature Distillation

3.2.1. PixelHop++

Recently, it is proved that the PixelHop++ system [30] can be used in object classification works by describing local block neighborhoods features of images. PixelHop++ contains two sub-units: neighborhood construction, c/w Saab (channel-wise subspace approximation via adjusted bias) transform. The neighborhood construction sub-unit is to compute attributes of near-to-far neighborhoods of selected pixels through all PixelHop++ units in multiple stages. The i-th Pixel-Hop++ unit concatenates attributes of the (i+1)-th neighborhood like a waterfall, where the i=1, 2, 3 in our work. The c/w Saab transform in PixelHop++ is a variant of principal component analysis (PCA) and it can decompose a signal space into the local mean and some frequency components by kernels. A kernel with a larger eigenvalue will extract a lower frequency component.

The purpose of using PixelHop++ is that it has very few parameters compared with CNN but achieves the feature extraction effect of CNN due to the neighborhood construction in PixelHop++ playing the equivalent role to convolutional filters in CNN.

3.2.2. Cascaded-Hop

Compared to the previous detector of DefakeHop, the novelties of our work were inspired by the EfficientNet, the image resolution and depth of CNN would affect the classification accuracy. We use the frames processed by RCAN (a super-resolution method) to recover high-frequency information of images by paying more attention to local features, which is corresponding to our method of PixelHop++ to extract local neighborhood features from high-resolution images. Like the CNN network, deepening the network depth can improve the classification accuracy to a certain extent. So, more features are extracted by the four cascaded PixelHop++. What's more, we use different sizes of kernels at different cascade layers to increase the model's expression ability. In the former two cascade stage, we apply 2×2 kernels to extract local neighborhood features. In the latter two cascade stages, we apply 3×3 kernels to extract global field features. Different sizes of kernels extract multiple types of receptive field feature vectors, which is consistent with the characteristic of Cascaded-Hop to extract local and global features from shallow to deep.

Our Cascaded-Hop method as shown in **Fig. 5** has four decomposed PixelHop++ units. The kernels' sizes in our four PixelHop++ units are different from the first hop to the fourth hop but with the same stride equal to one. We set the kernel size of the former two hops as 2×2 to retain more feature information in the initial stage and the latter two hops as 3×3 to avoid feature redundancy.

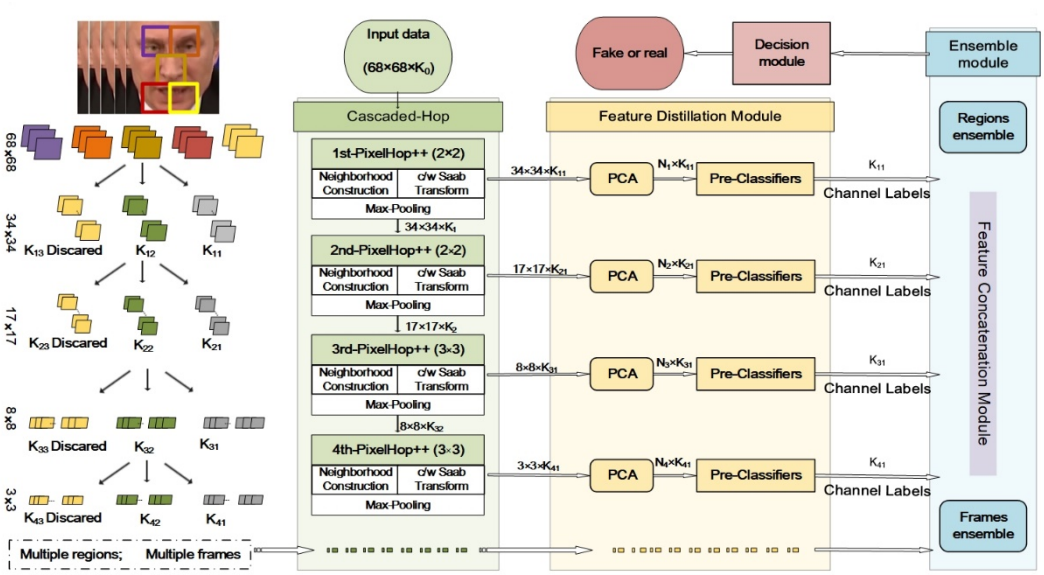


Fig. 3. The overview of our Cascaded-Hop model.

Take the first PixelHop++ unit as a sample, this PixelHop++ unit applies a 2×2 kernel to the input images. In consideration of the boundary effect, we applied (2×2)-to-(1×1) max-pooling. Thus, when using the colorful high-resolution images (68×68× K_0, where K_0 =3) as the input images, we would get K_1 (spectral dimension) local vectors of 2×2×K_0=12 spatial dimension, where K_1= K_{11}+K_{12}+ K_{13}.

Specifically, see the tree diagram on the left of **Fig. 5**, the input image, as the root node, generates the first-level child nodes. Once the color images begin sent into the Cascaded-Hop, the first PixelHop++ will yield a set of local vectors of dimension 12. We measure the frequency component by its channel energy and define the energy of the root node as one. Each child node energy in the tree, which represents a response of a certain frequency component, can be obtained and normalized from the root energy value.

According to their channel energy, those child nodes would be divided into three groups: high-energy channels (represent high-frequency components, yellow child nodes), mid-energy channels (represent mid-frequency components, green child nodes), and low-energy channels (represent low-frequency components, gray child nodes). Furtherly, we discard K_{13} high-energy channels due to their useless small eigenvalues. The mid-energy channels, whose dimension is 34×34×K_{11}, will be sent into the PCA module (to the right of the PixelHop++ unit shown in **Fig. 5**) to learn features furtherly. The K_{12} low-energy channels will be fed into the next PixelHop++ as the input data because low-frequency components are larger eigenvalues.

In terms of spectral dimension and spatial dimension analysis, our Cascaded-Hop system can be seen that the former hops contain more local features but a narrower view, the latter hops have less local features but a broader view. More it can be seen that our Cascaded-Hop model yielded different receptive fields.

3.2.3. Pre-classifier

To extract effective features to make the model has a good classification effect, we have two steps to process the features before the final decision module. Though local facial features can be produced from the Cascaded-Hop, those features received from PixelHop++ are not concise

enough to be used to classify whether the facial image has been manipulated or not due to their dimension being too large.

Especially in the first PixelHop++ unit as an example, the feature dimension will increase to $34\times34\times K_{11}\times n\times N$, where n is the number of regions of a face equal to 5, N is the number of frames of a video equal to dozens. Because of the strong spatial correlations for feature vectors from the same PixelHop++, we apply another PCA module to those vectors yielded from the former PixelHop++ units to reduce the dimension and weaken correlations between the spatial responses. Concretely speaking, we use the PCA module to select the top F_i components which contain about 90% of the images, where i=1, 2, 3, 4.

Through the above steps, we have kept the most effective facial features of the $F_i\times K_{i1}$ (where F is a feature channel vector's dimension, K is the number of the vector) dimension respectively from the i-th PixelHop++ and PCA models. To make efficient use of these features and reduce the possible misleading of individual vector channels to classification, we use XGBoost to build a pre-classifier, which can provide the probability of each channel related to the Deepfake video for the final classification module. For each PixelHop++, K_{i1} labels are sent to the ensemble module lastly.

3.3. Ensemble And Classification

After the subspace of multiple regions of multiple frames of a video is processed by the preprocessing method, PixelHop++ units, feature distillation module in order, pre-classifier labels as decision factors are sent into the aggregation module. As shown in the rightmost of **Fig. 5**, the ensemble module would ensemble different regions of a face and multiple frames of a video. We calculate the probability of forgery by averaging the probability of all labels from the same frame and video. Thus, both the frame-level probability and the video-level probability are figured out in the final decision.

4. Experiment

In this section, we divided the experiment into three subsections:

(1) Proving the effectiveness of our preprocessing method by a control experiment.
(2) Evaluating our Cascaded-Hop on multiple datasets to verify its performance.
(3) Comparing the AUC (area under curve) result of our methods with other solutions.

4.1. Dataset Benchmark

In this paper, AUC is used as the evaluation metric for the detector. AUC is commonly used to evaluate binary classification models, which is obtained by calculating the area of the ROC (receiver operating characteristic) curve. The closer AUC is to 1, the better the classification performance of the model is. There are two reasons for using AUC in this paper: On one hand, the number of positive and negative samples is unbalanced in the Celeb-DF dataset (thousands of fake videos, but few hundred true videos) while the AUC can objectively evaluate a model in the case of unbalanced sample categories. Another hand, the use of the AUC metric facilitates comparison with other Deepfake detectors using the same evaluation metric.

We have evaluated our Cascaded-Hop model on those most widely accepted Deepfake datasets: FaceForensics++ [8], Celeb-DF (v1) [50], Celeb-DF (v2) [52] and DFDC [53].

FaceForensics++ benchmark includes four types of video manipulation method subsets (DeepFake, Face2Face, FaceSwap, and NeuralTextures) and a pristine video subset, each of those five subsets is contains 1,000 videos. Two quality levels in FaceForensics++: high-

quality (FF++ (C23)) and low-quality (FF++ (C40)) videos are generated by constant rate quantization parameters equal to 23 and 40.

Celeb-DF (v1) benchmark contains only 795 Deepfake videos without pristine videos. Celeb-DF (v2) benchmark (is greatly extended from Celeb-DF (v1)) which includes 590 pristine videos and 5,639 Deepfake videos. In other words, the quality of the fake video in Celeb-DF (v2) is higher than fake videos in Celeb-DF (v1) because the manipulation techniques of Celeb-DF (v2) are improved greatly than Celeb-DF (v1).

DFDC benchmark offers a total of 5,244 videos of 1,131 real videos and 4,113 fake videos generated by two different manipulation methods. This is a challenging dataset.

After the preprocessing method is applied to those datasets, about 10% of both real videos and fake videos will be discarded because their quality is too low (the face is too blurry or the face size is too small to be suitable for our detector).

Therefore, the quantity of the detailed videos after preprocessing is shown in **Table 1**.

Table 1. The detailed number of videos processed by our method

Datasets	Subset	Original quantity	Saved quantity
FF++ (C23)	Pristine	1000	912
	DeepFake	1000	900
	Face2Face	1000	894
	FaceSwap	1000	917
	NeuralTextures	1000	899
FF++ (C40)	Pristine	1000	901
	DeepFake	1000	900
	Face2Face	1000	879
	FaceSwap	1000	886
	NeuralTextures	1000	881
Celeb-DF (v1)	DeepFake	795	738
Celeb-DF (v2)	Pristine	590	528
	DeepFake	5639	5070
DFDC	Pristine	1131	1068
	Fake	4113	3851

4.2. Experiments

Firstly, we verify the influence of the number of facial regions on the FF++ (C23) dataset. We selected about 60% of videos from the original subset and fake subset of FF++ (C23) as the training dataset, about 20% of the rest videos as the test dataset.

As the AUC result is shown in **Table 2**, with the increase of facial regions, the classification result is improved. We can conclude that the regions of the eyes, nose, and mouth have enough rich features for the detector to learn.

According to **Table 2**, this detector has the best performance on the DeepFake subset, and the worst detection result on NeuralTextures. This is because videos in DeepFake involve facial identity and attribute manipulation. But videos in NeuralTextures generally involve light reconstruction and texture information changes of the face.

Table 2. The AUC value of using different number of facial regions

Datasets	Level	Eyes	Eyes+Nose	Eyes+Nose+Mouth
DeepFake	Frame-level	0.7510	0.8324	0.9252
	Video-level	0.8285	0.8735	0.9613
Face2Face	Frame-level	0.7213	0.7951	0.9101
	Video-level	0.8340	0.8688	0.9527
FaccSwap	Frame-level	0.7662	0.8112	0.9055
	Video-level	0.8274	0.8589	0.9503
NeuralTextures	Frame-level	0.6687	0.7719	0.8170
	Video-level	0.7350	0.8127	0.8658

Secondly, we use the DeepFake (DF) of FF++ (C23) and DFDC to prove the effectiveness of our preprocessing method. We used 80% of DF and Original videos from FF++ (C23) for training and the rest 20% for testing. The same grouping method is applied to DFDC datasets. Then, frames are extracted in different ways: a) randomly selecting 40 face frames in the video; b) selecting frames by the steps of (1) and (2) but without (3) of our preprocessing method in **Fig. 3**; c) selecting and processing frames by using all the steps of our preprocessing method in **Fig. 3**.

The AUC result of the experiment is shown in **Table 3**. It can be seen that the Deepfake samples generated by our preprocessing method can make the Deepfake detector's classification effectiveness better, and the frame-level is improved by about 6%, and the video-level is improved by about 4%.

Table 3. Effectiveness of our preprocessing method

Datasets	Level	(a)	(b)	(c)
DF of FF++ (C23)	Frame-level	0.8730	0.8906	0.9252
	Video-level	0.9228	0.9371	0.9613
DFDC	Frame-level	0.8239	0.8357	0.8861
	Video-level	0.8745	0.8934	0.9201

Lastly, we have experimented with mixed datasets on FF++ (C23) and FF++ (C40) to evaluate the robustness of our Cascaded-Hop and explore whether the video quality will affect its performance. For the FF++ (C23) and FF++ (C40) datasets, different from the way we set up above, we selected 25% videos from each fake subset, mixing those videos as a mixed dataset. Selecting 80% of the mixed dataset for training, and another 20% for testing. As for Celeb-DF (v1) and Celeb-DF (v2), we take the real dataset of Celeb-DF (v2) as the shared dataset between Celeb-DF (v1) and Celeb-DF (v2), 70% for training and 30% for testing. However, the fake datasets of the two Celeb-DF datasets are used separately for training and testing.

The AUC value of experiments is shown in **Table 4**. We can find that the performance of the Cascaded-Hop will be different on different quality videos caused of compression factors. According to the analysis of these experimental results, Cascaded-Hop's discrimination ability at both frame-level and video level decreases in the mixed datasets, and the comparison is more obvious in FF++. It means that different face manipulation types have different feature spatial distributions, which has an impact on the detector to learning features. What's more,

Celeb-DF (v2) contained more complex video data sources than Celeb-DF (v1), and the AUC result on Celeb-DF (v2) is worse than Celeb-DF (v1).

Table 4. The AUC result on mixed datasets

Datasets	Frame-level	Video-level
FF++ (C23)	0.8945	0.9379
FF++ (C40)	0.8530	0.9115
Celeb-DF (v1)	0.8870	0.9225
Celeb-DF (v2)	0.8321	0.8917

4.3. Evaluation

We compare our Cascaded-Hop to other Deepfake detection studies [1, 8, 10] and detectors [12, 13, 15, 25, 26, 41]. These works all use AUC to evaluate the performance of detection models and evaluated on the DF of FF++ (C23), DFDC, and Celeb-DF (v2) datasets. These AUC results were extracted from the studies of [1, 8, 10] or the original publications of [12, 13, 15, 25, 26, 41]. Blank cells indicate that the relevant data was not found.

As shown in **Table 5**, it can be seen that most detectors show good identification results on DF of FF++ (C23), and our Cascaded-Hop has an excellent performance, too. Our detector performed better on DFDC and Celeb-DF (v2) than others which means Cascaded-Hop's nice classification performance on multiple datasets. Especially on Celeb-DF (v2), our model detection effect is much better than other works. It is worth mentioning that the AUC result of our model is slightly worse than DefakeHop. One possible reason is that in the work of [41], the authors randomly select 80% for training and 20% for testing from the datasets, which means that some test videos may overlap with training videos.

Table 5. Compare with other methods

Detectors	DF of FF++ (C23)	DFDC	Celeb-DF (v2)
Inception V3 [1]	70.1%	--	55.7%
Xception-raw [8]	99.7%	49.9%	48.2%
Xception [10]	--	91.1%	83.6%
Meso4 [12]	84.7%	75.3%	54.8%
MesoInception4 [12]	83.0%	73.2%	53.6%
Multi-task [13]	76.3%	53.6%	54.3%
DenseNet+RNN [15]	96.9%	--	--
HeadPose [25]	47.3%	55.9%	54.6%
DSP-FWA [26]	93.0%	75.5%	64.6%
DefakeHop [41]	97.45%	--	90.56%
Cascaded-Hop	96.13%	92.01%	89.17%

5. Conclusion

In this paper, we provide a method to extract facial image frame sequences from videos, which can make high-quality Deepfake samples datasets for training and testing detectors. Importantly, we design and implement a detector named Cascaded-Hop based on SSL and PixelHop++. This Deepfake videos detector performs better classification results than other models on multiple public Deepfake datasets, especially on the Celeb-DF (v2) dataset.

However, some disadvantages of Cascaded-Hop are that when the image is large enough, the cropped subspace size may only cover a relatively small region, and our model cannot capture the temporal information like detectors based on the deep learning network.

6. References

[1] R. Tolosana, R. Vera-Rodriguez, J. Fierrez, A. Morales, and J. Ortega-Garcia, "Deepfakes and beyond: A survey of face manipulation and fake detection," *Information Fusion*, vol. 64, pp. 131-148, Dec. 2020. Article (CrossRef Link)

[2] M. Zhang, K. Zeng, and J. Wang, "A survey on face anti-spoofing algorithms," *Journal of Information Hiding and Privacy Protection*, vol. 2, no. 1, pp. 21–34, Oct. 2020. Article (CrossRef Link)

[3] S. Suwajanakorn, S. Seitz, and I. Kemelmacher-Shlizerman, "Synthesizing Obama: Learning Lip Sync From Audio," *ACM Transactions on Graphics*, vol. 36, no. 4, pp. 1–13, Aug. 2017. Article (CrossRef Link)

[4] S. Agarwal, H. Farid, Y. Gu, M. He, K. Nagano et al., "Protecting world leaders against deep fakes," in *Proc. of CVPR*, Long Beach, USA, pp. 38–45, 2019.

[5] B. Hu, J. Wang, "Deep learning for distinguishing computer generated images and natural images: a survey," *Journal of Information Hiding and Privacy Protection*, vol. 2, no.2, pp. 95–105, Nov. 2020. Article (CrossRef Link)

[6] P. Korus, "Digital image integrity: a survey of protection and verification techniques Digit," *Digital Signal Processing*, vol. 71, pp. 1-26, Dec. 2017. Article (CrossRef Link)

[7] A. Piva, "An overview on image forensics," *ISRN Signal Processing*, vol. 2013, pp. 1-22, Jan. 2013. Article (CrossRef Link)

[8] A. Rössler, D. Cozzolino, L. Verdoliva, C. Riess, J. Thies, M. Nießner, "FaceForensics++: Learning to Detect Manipulated Facial Images," in *Proc. of ICCV*, Seoul, Korea (South), pp. 1-11, 2019. Article (CrossRef Link)

[9] D. Wodajo, and S. Atnafu, "Deepfake Video Detection Using Convolutional Vision Transformer," *Eprint arXiv: 2102.11126*, 2021. Article (CrossRef Link)

[10] R. Tolosana, S. Romero-Tapiador, J. Fierrez, and R. Vera-Rodriguez, "DeepFake evolution: Analysis of facial regions and fake detection performance," in *Proc. of ICPR*, Sanya, China, pp. 442-456, 2021. Article (CrossRef Link)

[11] P. Zhou, X. Han, V. I Morariu, and L. S. Davis, "Two-stream neural networks for tampered face detection," in *Proc. of CVPR*, Hawaii, USA, pp. 1831–1839, 2017. Article (CrossRef Link)

[12] D. Afchar, V. Nozick, J. Yamagishi, and I. Echizen, "Mesonet: a compact facial video forgery detection network," in *Proc. of WIFS*, Hong Kong, China, pp. 1–7, 2018. Article (CrossRef Link)

[13] H. H. Nguyen, F. Fang, J. Yamagishi, and I. Echizen, "Multi-task learning for detecting and segmenting manipulated facial images and videos," in *Proc. of 2019 IEEE 10th International Conference on Biometrics Theory, Applications and Systems (BTAS)*, 2019. Article (CrossRef Link)

[14] Y. Li and S. Lyu, "Exposing DeepFake Videos By Detecting Face Warping Artifacts," *Eprint arXiv:1811.00656*, 2018. Article (CrossRef Link)

[15] E. Sabir, J. Cheng, A. Jaiswal, W. AbdAlmageed, I. Masi et al., "Recurrent convolutional strategies for face manipulation detection in videos," in *Proc. of CVPR*, Long Beach, USA, pp. 80–87, 2019. Article (CrossRef Link)

[16] D. M. Montserrat, H. Hao, S. K. Yarlagadda, S. Baireddy, R. Shao et al., "DeepFake Detection with Automatic Face Weighting," in *Proc. of CVPRW*, Salt Lake City, Utah, USA, pp. 668-669, 2020. Article (CrossRef Link)

[17] D. Güera, and E. J. Delp, "Deepfake video detection using recurrent neural networks," in *Proc. of AVSS*, Auckland, New Zealand, pp. 1–6, 2018. Article (CrossRef Link)

[18] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu et al., "Generative Adversarial Nets," in *Proc. of NIPS*, Long Beach, USA, pp. 2672–2680, 2014. Article (CrossRef Link)

[19] C. L. Wang, Y. L. Liu, Y. J. Tong, J. W. Wang, "GAN-GLS: Generative Lyric Steganography based on Generative Adversarial Networks," *Computers, Materials & Continua*, vol. 69, no.1, pp. 1375-1390, Jun. 2021. Article (CrossRef Link)
[20] J. Donahue, L. Anne Hendricks, S. Guadarrama, M. Rohrbach, S. Venugopalan et al., "Long-Term Recurrent Convolutional Networks for Visual Recognition and Description," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 4, pp. 677-691, 2017. Article (CrossRef Link)
[21] H. Li, B. Li, S. Tan, and J. Huang, "Detection of deep network generated images using disparities in color components," *Eprint arXiv:1808.07276*, 2018. Article (CrossRef Link)
[22] S. McCloskey, and M. Albright, "Detecting GAN-generated Imagery using Color Cues," *Eprint arXiv:1812.08247*, 2018. Article (CrossRef Link)
[23] J. C. Neves, R. Tolosana, R. Vera-Rodriguez, V. Lopes, H. Proença et al., "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, Jul. 2020. Article (CrossRef Link)
[24] N. Yu, L. Davis and M. Fritz, "Attributing Fake Images to GANs: Learning and Analyzing GAN Fingerprints," in *Proc. of ICCV*, Santiago, Chile, pp. 7556-7566, 2019. Article (CrossRef Link)
[25] X. Yang, Y. Li, and S. Lyu, "Exposing DeepFake using inconsistent head poses," in *Proc. of ICASSP*, Brighton, UK, pp. 8261–8265, 2019. Article (CrossRef Link)
[26] F. Matern, C. Riess, and M. Stamminger, "Exploiting visual artifacts to expose DeepFake and face manipulations," in *Proc. of WACVW*, California, USA, pp. 83–92, 2019. Article (CrossRef Link)
[27] L. Li, J. Bao, T. Zhang, H. Yang, D. Chen, F. Wen et al., "Face X-ray for More General Face Forgery Detection," in *Proc. of CVPR*, Long Beach, USA, pp. 5000-5009, 2019. Article (CrossRef Link)
[28] P. Yu, J. Fei, Z. Xia, Z. Zhou and J. Weng, "Improving Generalization by Commonality Learning in Face Forgery Detection," *IEEE Transactions on Information Forensics and Security*, vol. 17, pp. 547-558, Jan. 2022. Article (CrossRef Link)
[29] Y. Zhang, K. Li, K. Li, L. Wang, B. Zhong et al., "Image Super-Resolution Using Very Deep Residual Channel Attention Networks," in *Proc. of ECCV*, Munich, Germany, pp. 294-310, 2018. Article (CrossRef Link)
[30] Y. Chen, M. Rouhsedaghat, S. You, R. Rao, and C.-C. J. Kuo, "Pixelhop++: A small successive-subspace-learning-based (SSL-based) model for image classification," in *Proc. of ICIP*, Abu Dhabi, United Arab Emirates, pp. 3294-3298, 2020. Article (CrossRef Link)
[31] C.-C. Jay Kuo, "Understanding convolutional neural networks with a mathematical model," *Journal of Visual Communication and Image Representation*, vol. 41, pp. 406–413, Nov. 2016. Article (CrossRef Link)
[32] C.-C. J. Kuo, M. Zhang, S. Li, J. Duan, and Y. Chen, "Interpretable convolutional neural networks via feedforward design," *Journal of Visual Communication and Image Representation*, vol. 60, pp. 346–359, Apr. 2019. Article (CrossRef Link)
[33] M. Rouhsedaghat, Y. Wang, X. Ge, S. Hu, S. You, "Facehop: A light-weight low-resolution face gender classification method," in *Proc. of ICPR*, Hong Kong, China, pp. 169-183, 2021. Article (CrossRef Link)
[34] Deepfake project-non-official project based on original DeepFake thread [Online]. Available: https://github.com/Madhivarman/deepfake_faceswap, Accessed on: Jan. 2, 2018,
[35] P. Korshunov and S. Marcel, "DeepFake: a New Threat to Face Recognition? Assessment and Detection," *Eprint arXiv:1812.08685*, 2018. Article (CrossRef Link)
[36] T. Karras, S. Laine and T. Aila, "A Style-Based Generator Architecture for Generative Adversarial Networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, no. 12, pp. 4217-4228, Dec. 2021. Article (CrossRef Link)
[37] Y. Choi, M. Choi, M. Kim, J. Ha, S. Kim et al., "StarGAN: Unified Generative Adversarial Networks for Multi-Domain Image-to-Image Translation," in *Proc. of CVPR*, Salt Lake City, UT, USA, pp. 8789-8797, 2018. Article (CrossRef Link)

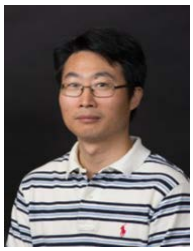
[38] A. Bansal, S. Ma, D. Ramanan, and Y. Sheikh, "Recycle-GAN: Unsupervised video retargeting," in *Proc. of ECCV*, Munich, Germany, pp. 119-135, 2018. Article (CrossRef Link)
[39] J. Thies, M. Zollhofer, M. Stamminger, C. Theobalt, and M. Nießner, "Face2face: Real-Time Face Capture and Reenactment of RGB Videos," in *Proc. of CVPR*, Las Vegas, USA, pp. 2387-2395, 2016. Article (CrossRef Link)
[40] Y.Z. Li, B.Y. Sun, T. F. Wu, Y. Wang, "Face Detection with End-to-End Integration of a ConvNet and a 3D Model," in *Proc. of ECCV*, Amsterdam, Netherlands, pp. 420-436, 2016. Article (CrossRef Link)
[41] H.S. Chen, M. Rouhsedaghat, H. Ghani, et al., "Defakehop: A light-weight high-performance deepfake detector," in *Proc. of ICME*. Shenzhen, China, pp. 1-6, 2021. Article (CrossRef Link)
[42] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov et al., "Going deeper with convolutions," in *Proc. of CVPR*, Boston, MA, USA, pp. 1-9, 2015.
[43] D. Y. Zhang, J. W. Hu, F. Li, X. L. Ding, A. K. Sangaiah, V. S. Sheng, "Small Object Detection via Precise Region-Based Fully Convolutional Networks," *Computers, Materials & Continua*, vol. 69, no. 2, pp. 1503-1517, Jul. 2021. Article (CrossRef Link)
[44] H. P. Wu, Y. L. Liu, J. W. Wang, "Review of Text Classification Methods on Deep Learning," *Computers, Materials & Continua*, vol. 63, no. 3, pp. 1309-1321, Apr. 2020. Article (CrossRef Link)
[45] T. Shahroz, L. Sangyup, K. Hoyoung, S. Youjin, and S.W. Simon, "Detecting Both Machine and Human Created Fake Face Images In the Wild," in *Proc. of MPS*, Toronto, Canada, pp. 81-87, 2018. Article (CrossRef Link)
[46] H. H Nguyen, F. Fang, J. Yamagishi, and I. Echizen, "Multi-task learning for detecting and segmenting manipulated facial images and videos," in *Proc. of BTAS*, Tampa, FL, USA, pp. 1-8, 2019. Article (CrossRef Link)
[47] H. H Nguyen, J Yamagishi, and I. Echizen, "Use of a capsule network to detect fake images and videos," *Eprint arXiv:1910.12467*, 2019. Article (CrossRef Link)
[48] Y. Li, M. Chang, S. Lyu, "In Ictu Oculi: Exposing AI Generated Fake Face Videos by Detecting Eye Blinking," in *Proc. of WIFS*, Hong Kong, China, pp. 1-7, 2018. Article (CrossRef Link)
[49] H. Li, P. He, S. Wang, A. Rocha, X. Jiang et al., "Learning generalized deep feature representation for face anti-spoofing," *IEEE Transactions on Information Forensics and Security*, vol. 13, no. 10, pp. 2639–2652, Oct. 2018. Article (CrossRef Link)
[50] Y. Li, X. Yang, P. Sun, H. Qi and S. Lyu, "Celeb-DF: A Large-scale Challenging Dataset for DeepFake Forensics," in *Proc. of CVPR*, Seattle, WA, USA, pp. 3204-3213, 2020. Article (CrossRef Link)
[51] C. R. Wang and W. H. Deng, "Representative Forgery Mining for Fake Face Detection," in *Proc. of CVPR*, Nashville, TN, USA, pp. 14918-14927, 2021. Article (CrossRef Link)
[52] Y. Li, X. Yang, P. Sun, H. Qi and S. Lyu, "Celeb-DF(v2): A new dataset for Deepfake forensics," *Eprint arXiv:1909.12962*, 2019. Article (CrossRef Link)
[53] B. Dolhansky, R. Howes, B. Pflaum, N. Baram, and C. C. Ferrer, "The deepfake detection challenge (DFDC) preview dataset," *Eprint arXiv:1910.08854*, 2019. Article (CrossRef Link)

Dengyong Zhang received the B.S. and M.S. degree from Changsha University of Science and Technology, Changsha, China, in 2003, 2006 respectively. He received Ph.D. degree from Hunan University, China, in 2018. Now, He is an associate professor at Changsha University of Science and Technology. His current research interests include digital media forensics and video passive forensics

Pengjie Wu received the B.S. degree from Wuyi University, Nanping, China, in 2019. He is a postgraduate student in Changsha University of Science and Technology, Changsha, China. His research interests include computer vision, digital multimedia forensics, and deep learning.

Feng Li received the B.S. degree from Hunan Normal University, China, in 1984. He received M.S. degree from Zhejiang University, China, in 1988. He received Ph.D. degree from Sun Yat-sen University, China, in 2003. He is a professor at Changsha University of Science and Technology. His main research interests lie in the areas of human pose estimation, computer vision, pattern recognition and information security.

Wenjie Zhu received the B.S. degree from Xuzhou Institute of Technology, Xuzhou, China, in 2020. He is a postgraduate student in Changsha University of Science and Technology, Changsha, China. His research interests include computer vision, digital multimedia forensics, and deep learning.



Victor S. Sheng is an Associate Professor of Computer Science at Texas Tech University, and the founding Director of Data Analytics Lab (DAL). He received his Master's degree in Computer Science from the University of New Brunswick, Canada, in 2003, and his Ph.D. degree in Computer Science from Western University, Ontario, Canada, in 2007. His research interests include data mining, machine learning, crowdsourcing, and related applications in business, industry, medical informatics, and software engineering.