SDHF: Spotting DeepFakes with Hierarchical Features

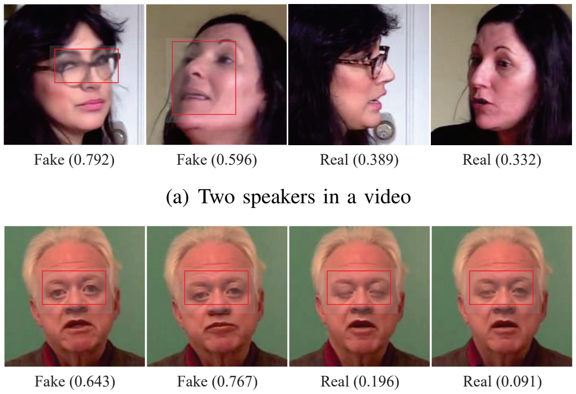
Tao Liang*†, Peng Chen*†, Guangzhi Zhou*†, Hongchao Gao*†, Jin Liu*, Zhaoxing Li*, Jiao Dai*
*Institute of Information Engineering, Chinese Academy of Sciences, Beijing, China
†School of Cyber Security, University of Chinese Academy of Sciences, Beijing, China
{liangtao0305, chenpeng, zhouguangzhi, gaohongchao, liujin, lizhaoxing, daijiao}@iie.ac.cn

***Abstract*—DeepFake videos are widely distributed on social media platforms, which has seriously affected the authenticity of digital media content, calling for robust DeepFake detection methods. Although numerous detection methods are formulated as frame-based binary classification, less attention has been paid to aggregate the features over individual frames to get a video-based judgement. We observed that for the detection of DeepFake videos, three different level forgery features from frame, clip and video can complement each other. We also found that discrete, large interval sampling strategy is more suitable for DeepFake detection, which can sample more complex video scenes, including multiple subjects, diverse facial expressions and head poses. In this work, we propose a hierarchical framework, using 2D convolutional neural networks for frame-level features extraction followed by a 1D convolutional aggregator to extract clip-level and video-level features, which can comprehensively exploit three different levels of features to make decisions. Our approach is easy to extend and can be applied to any frame-based detection model. Evaluation was performed on four datasets, including DFDC, Celeb-DF, FaceForensics++ and UADFV, which provides competitive results compared to other methods. Experimental results of cross-test demonstrate that our hierarchical framework has excellent generalization performance in the face of unknown datasets.**

***Index Terms*—Facial manipulation, DeepFake Detection, Hierarchical**

I. Introduction

In recent years, facial manipulation videos, also widely known as DeepFakes, have been increasing on social media platforms. Especially, huge steps forward in variational auto-encoders (VAEs) [1], generative adversarial networks (GANs) [2], [3] and computer graphics (CGs) based methods [4], [5] have made facial image manipulations reach photo-realistic level. Although such technologies greatly enrich people's entertainment life, it also seriously threatens the credibility of digital media content. Through face swapping, an individual can be placed at some location he or she was never present at. By changing lip movements and related speech signals, real video can be generated that allows people to say words they have never actually said. When this technology is deliberately used by criminals, it seriously affect the security of public opinion. In addition, as one of the most important biological signals, face is widely used in various applications, such as face payment and face access control. The forged face becomes a key to open our privacy. All mentioned above are calling for robust facial manipulation detection methods.



(a) Two speakers in a video

(b) Difficult samples due to blurry faces

Fig. 1. Some difficulties encountered in the process of integrating frame-based results to get video-based results: 1(a) a fake video containing multiple subjects' faces, in which only one or a few faces are manipulated for a fraction of the frames. 1(b) a real video with some highly blurred or compressed frames, lead the model to judge it as fake. The label under the picture indicates the probability of face being manipulated reported by EfficientNetB0 trained on DFDC. (Zoom in to see better)

To spot such DeepFake videos, significant progress has been made by handcrafted features [6]–[8] and deep learning [9]–[11]. [6] proposed to expose DeepFake videos by detecting the rate of eye blinking. [10] introduced two CNN based architectures for manipulation detection, which combines information from both low level (microscopic) and high level (macroscopic) features. [12] introduced segmentation tasks into manipulated face detection and tried to expose forged regions. The predominant efforts are focused on still images, less attention has been paid to aggregate the features over individual frames to get a video-based judgement. In practical applications, we need to make a judgment on the authenticity of a video. In order to aggregate the predictions over individual frames to get a video-based judgement, a natural approach is to average the predictions across all frames. However, this approach has several drawbacks and encounters many difficulties as shown in Figure 1. First, a fake video containing multiple subjects' faces, in which only one or a few faces are manipulated for a fraction of the frames. Second, some highly blurred or compressed frames in a real video will lead the model to judge it as fake. Also, some frames might contain blurry faces where the presence of manipulations might be difficult to detect. Furthermore, face detectors such as Retinaface [13] can erroneously report background regions of the frames with faces, resulting false positives. In such complicated scenarios, a model could provide a correct prediction for each frame but an incorrect video-based prediction after averaging.

Meanwhile, some previous clip-based works tried to leverage RNN [14] or optical flow [15] to capture manipulation traces in temporal, which demand a small interval between frames in the clip. These methods are robust compared to averaging all frames in obtaining video-based prediction. Nevertheless, such methods rely on carefully processed data and small frame interval to effectively capture traces in temporal. When encountering the complex situation mentioned above, the effectiveness of such methods drop sharply. Hence, it's sub-optimal to directly introduce RNN or optical flow for video-based judgment and these methods are hard to be deployed in real systems. In fact, the ideas of these clip-based methods are borrowed from action recognition, but we argue there are some differences between DeepFake video detection and action recognition. First, the manipulated face on the video usually only occupies a small area on the frame, so most detection methods first perform face detection and crop out the face to represent this frame. Factors such as inaccuracy of face detection, multiple subjects and movement of the face will seriously affect the capture of temporal information after face cropping. Second, The actions in action recognition are mostly rigid movements, while the tampered areas in the fake videos are mostly non-rigid movements, such as expression changes. Rigid motion is more easily captured by optical flow or RNN. Third, the characteristics of action recognition rely on multiple consecutive whole frames without any preprocess. Traces of video forgery may exist in frames, clips and videos at the same time, so we can synthesize three different levels of features to make a comprehensive decision.

Therefore, in this work, we propose to use discrete, large interval sampling strategy, which can sample diverse facial expressions and head poses, and sample the complex scenes mentioned above as much as possible, so as to make a comprehensive decision for a video. we propose a novel hierarchical framework named **SDHF**, using 2D convolutional neural networks for frame-level features extraction followed by a 1D convolutional aggregator to extract clip-level and video-level features, which can comprehensively exploit three different levels of features to make decision. In experiments, we adopt EfficientNet [16] as the frame-level feature extractor. And our approach is very convenient to extend and can applied to any frame-based DeepFake video detection model. Our SDHF follows a two-stage training process. First, the EfficientNet is trained on the training frames to extract features to discern real faces from synthetics. Second, we leverage the previously trained classifier to separately extract multi-frame features, then stack these features and send it into a 1D convolutional aggregator to extract clip-level and video-level features respectively. Finally, we synthesize the three level features to make a comprehensive decision. Especially when the real frames are mixed with a manipulated video, our approach can still confidently judge it as fake. The contributions of our work are summarized as follows:

(1) we propose a novel hierarchical framework **SDHF** for DeepFake video detection, which combines three different levels of features, including frame-level, clip-level and video-level, to comprehensively determine whether a video is manipulated.

(2) We observed that discrete, large interval sampling strategy is more suitable for DeepFake video detection, which can sample more comprehensive video scenes, including multiple subjects, diverse facial expressions and head poses. Correspondingly, we propose to apply the 1D convolutional module as an aggregator to automatically extract typical forged features present in the video.

(3) We evaluate the performance of the proposed SDHF on four datasets: DFDC [17], Celeb-DF [18], FaceForensics++ [9] and UADFV [7]. The results of intra-test demonstrate that our approach achieves better performance compared to other methods and boosts balanced accuracy with minimal dependency on the number of frames required per video. The results of cross-test prove that our hierarchical framework has excellent generalization performance in the face of unknown datasets.

II. Related Work

A. Facial Manipulation

Over the past decade, interest on facial manipulation research has increased a lot. Facial manipulation methods mainly fall into face replacement and face reenactment [9], while the former transforms the identity information from one to another and the latter focuses on motion transfer. Early works [19]–[21] mainly rely on graphics-based methods, using face models from sources, either warping or projecting on the target ones. Face2Face [4] is a real-time face reenactment system using only an RGB stream. It selects frames from each video to create dense reconstruction of the two faces. These dense reconstructions are used to re-synthesize the target face with different expressions under different lighting conditions. In Deepfakes [22], two auto-encoders (with a shared encoder) are trained to reconstruct target and source faces. Recently, VAEs [1] and GANs [2] are successfully employed to make forged videos more sophisticated. Traditionally, manipulation videos directly coming from generated face images lack temporal smoothness. With the popularity of facial manipulation research, works on face occlusion [23] or face motion representation [24] are focused on improving the quality of generated images or videos. [25] uses optical flow to improve smoothness between two adjacent frames while [26] applies 3D convolution to judge the realness of three consecutive frames.

The above two types of facial manipulation methods all share the same typical pipeline, as shown in Figure 2. The generation of facial manipulation video is usually performed frame by frame, which strongly depends on the performance

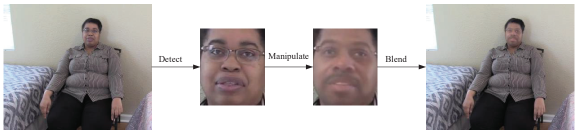
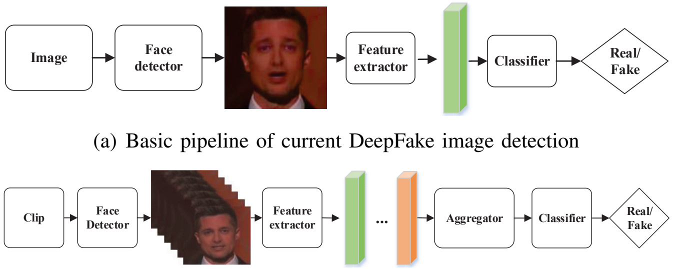


Fig. 2. Overview of a typical video facial manipulation pipeline.

of the face detector. Therefore, in a fake video, it is inevitable that part of the original real face will be retained due to failure of detection. Especially when more than one subjects' face appear in the video, the manipulation process will become quite chaotic and leave obvious visual traces.

B. DeepFake Video Detection

Despite the great development of facial manipulation technology, detection methods are still far behind. Previous works on DeepFake video detection fall into two categories: frame-based [27]–[30] and clip-based [6], [8], [14], [31], [32]. The former treat video as a frame set and convert the problem to detect fake faces in a single frame, whereas the latter treat video as a clip set attempt to model the temporal inconsistency between frames. Now we propose a video-based framework.



(a) Basic pipeline of current DeepFake image detection

(b) Basic pipeline of current DeepFake clip detection

Fig. 3. Basic pipeline of current DeepFake image/clip detection

Frame-based. As shown in Figure 3(a), the current DeepFake image detection approaches share the same pipeline. The main innovations of different methods focus on the feature extractor. Several algorithms using handcrafted features, deep learning algorithms and lately GAN-based methods are being explored. [33] utilized 3D head pose inconsistencies to distinguish real and fake videos. In MesoNet [10], shallow architectures with an inclusion of inception module learns the discriminative feature from frames. On the other hand, recent works [34] have proved that deep architectures outperform the shallow networks by a large margin. [27] proposed a two-stream CNN for tampered face detection, combing RGB space features with steganalysis features. This is a combination of deep learning features and handcrafted features. In the multi-task learning approach, classification, segmentation [35], and reconstruction [36] is performed altogether to boost classification performance. However, all of these methods only consider spatial information in a single frame. In order to get a video-based judgment, a natural approach is to average the predictions across all frames. This is obviously a sub-optimal solution, which has several drawbacks and encounters many difficulties as shown in Figure 1.

Clip-based. As shown in Figure 3(b), the current DeepFake clip detection approaches share the same pipeline. The main differences between various methods are the feature extractor and aggregator. Some clip-based approaches sought for physiological signals as features, like eye blinking [6], and rPPG [8], but quickly lost its effect when facing more training data and advanced generative models. [31] introduced RNN as the aggregator to capture temporal inconsistencies between frames. It is a two-stage strategy composed of a CNN to extract frame-level features followed by an RNN aggregator. [14] extended this work by a bidirectional RNN and train the model end-to-end. The above methods are very sensitive to the length of the clip, but the length of the test data in the real world is unknown. The core idea of these two methods [14], [31] are borrowed from action recognition approaches. However, due to the difference between action recognition and DeepFake detection, it's sub-optimal to directly use the method of action recognition. Although these methods are robust compared to averaging all frames in obtaining video-based prediction, such methods rely on carefully processed data to effectively capture features in temporal.

Video-based. To our knowledge, we are the first to propose a video-based framework for DeepFake detection other than averaging across all frames/clips. DeepFake videos have their inherent characteristics. On one hand, Each frame and each sparse clip can be used as a criterion, and action recognition must take temporal continuous clip as a criterion. On the other hand, discrete, large interval sampling strategy is more suitable for DeepFake video detection, which can sample more comprehensive video scenes, including multiple subjects, diverse facial expressions and head poses. Based on these two observations, we propose to utilize 2D convolutional neural networks to extract frame-level manipulation traces, followed by an aggregator composed of multiple 1D convolutional blocks to capture traces existed in clips and videos. Combining multi-level features, we can make an accurate judgment on a video. Last but not least, our approach is robust to video length.

III. METHODOLOGY

The goal of our framework, **SDHF**, is to extract hierarchical features in DeepFake videos, including frame-level, clip-level and video-level, and synthesize three different levels of features to comprehensively determine whether the video is manipulated. Our SDHF also follows the structure shown in Figure 3(b). In the following, we first introduce the data processing strategy and the details of 1D convolutional block. Then, our proposed framework, loss functions and network training strategy are described respectively. Figure 4 presents the whole framework.

A. Data Processing Strategy

For a given video, our purpose is to determine whether there are manipulations at a minimum cost. Initially, we extract 16

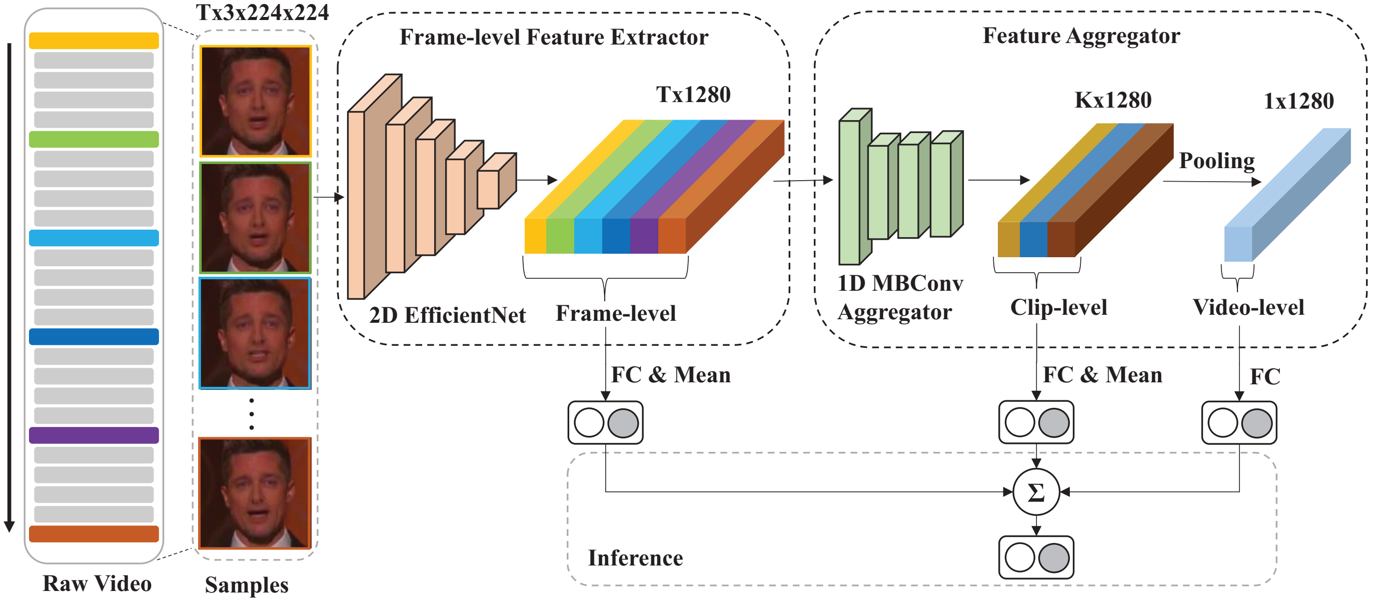


Fig. 4. The overall framework of SDHF. There are two stages in this framework. First, the EfficientNet is trained on the training frames minimizing cross-entropy loss for binary classification to extract features to discern real faces from synthetics. Second, use the previously trained classifier to separately extract multi-frame features, then stack these features and send it into a 1D convolutional aggregator to extract clip-level and video-level features respectively.

frames uniformly from a given video and the frame interval is relatively large and discrete. Because all visual manipulations are located within face regions, and faces are typically present in a small region of the frame, we focus on extracting traces only in regions where a face is present. Although some datasets provide face masks, which can be used for accurate face positioning and cropping, we cannot obtain masks in online DeepFake detection scenarios. Therefore, we try to simulate real business scenarios and use a pretrained Retinaface [13] for face detection. Then, we get the minimal square box by warping the bounding box. Finally, the box is enlarged by 1.3 times as cropped region, where not only the core areas of face but also sufficient surrounding areas are covered.

B. 1D MBConv block

Variations and replacements for 2D convolutional blocks have sprung up, which mainly use neural architecture search (NAS) algorithms to find a balance between efficiency and accuracy. With the successful experience of 2D convolutional block in MobileNets [38], we generalize it to 1D convolutional to get the basic block, as shown in Figure 5. Its main structure is mobile inverted bottleneck MBConv [38], [39] with squeeze-and-excitation (SE) [37] optimization added. Note that we use pointwise convolutional in the SE attention structure (Figure 5(c)) instead of full connection. In addition, both drop connection [40] and residual connection [41] is adopted to prevent overfitting.

C. Framework

Frame-level Feature Extractor. After cropping face regions, a binary classifier is trained to extract features that can be used to classify the real/fake faces. Among the family of EfficientNets [16], we choose EfficientNetB0 as the baseline in our work for fair comparisons. Additionally, the network has been designed using neural architecture search (NAS) algorithms, resulting in a network EfficientNetB0 that is both compact and accurate. In fact, this network has outperformed previous state-of-the-art approaches in datasets such as ImageNet [42] while having fewer parameters. More details about EfficientNetB0 can be found in [16].

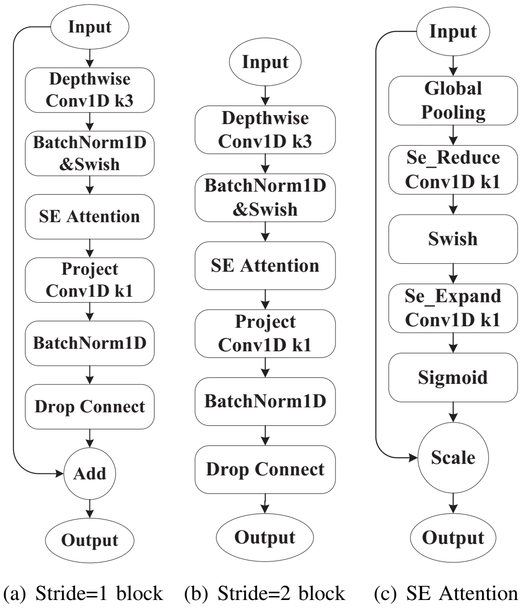


Fig. 5. Illustration of 1D convolutional blocks used in our SDHF. 5(c) The attention mechanism is modified from SeNet [37] to learn importance of difference channels.

This module takes videos as inputs $\mathbf{X} \in \mathbb{R}^{B \times T \times C \times H \times W}$ and generates the frame-level features $\mathbf{F}_{frame} \in \mathbb{R}^{B \times T \times C \times H \times W}$ and the probability of each frame being fake $logits_{frame}$,

$$\mathbf{F}_{frame}, logits_{frame} = EfficientNet(X) \tag{1}$$

where B, T, C, H and W, are batch size, number of frames, number of channels, height and width, respectively. Then we adopt spatial global average pooling to get the 1D frame-level features $\mathbf{F'}_{frame} \in \mathbb{R}^{B \times T \times C}$ for the analysis of next stage. Of course, we argue that better results can be obtained if EfficientNetB1-B7 is used as the backbone.

Feature Aggregator. In the previous step we have obtained the frame-based predictions $logits_{frame}$, the simplest choice is to average the predictions across all frames to obtain a video-based prediction. However, this approach has several drawbacks and encounters many difficulties as shown in Figure 1. First, some fake videos containing multiple subjects' faces, in which only one or a few faces are manipulated for a fraction of the frames. Second, some real videos with some of the frames are highly blurred or compressed, leading the model to judge it as fake. Also, some frames might contain blurry faces where the presence of manipulations might be difficult to detect. Furthermore, face detectors such as Retinaface [13] can erroneously report background regions of the frames contain faces, resulting false positives. In such complicated scenarios, a model could provide a correct prediction for each frame but an incorrect video-based prediction after averaging. In order to address this problem, we propose to adopt 1D convolution as aggregator, which can automatically extract typical features from $\mathbf{F'}_{frame}$ as criteria.

TABLE I
1D MBConv Aggregator

Stage i	Operator $\hat{\mathcal{F}}_i$	Resolution $\hat{T}_i$	#Channels $\hat{C}_i$	#Layers $\hat{\mathcal{L}}_i$
1	1D-MBConv,k=3,s=2	16	1280	1
2	1D-MBConv,k=3,s=2	8	1280	4
3	Pooling	8	1280	1

The feature aggregator is composed of multiple 1D MBConv [38], [39] blocks described in Figure 5. The aggregator details are shown in Table I. In the first stage, we exploit a 1D MBConv block with a step size of 2 to halve the video multi-frame feature map. In the second stage, we stack 4 1D MBConv blocks with the same step size of 1 to refine the feature map. All the above 1D MBConv blocks maintain the same feature dimension. This module takes the transposed frame-level features as input $\mathbf{F'}^{T}_{frame}$ and generates clip-level features $\mathbf{F}_{clip} \in \mathbb{R}^{B \times C \times T'}$. Then, the clip-level features are squeezed into video-level features $\mathbf{F}_{video} \in \mathbb{R}^{B \times C}$ by an average pooling along T dimension. Finally, three different level features are independently sent to the fully connected layers for classification, and we weight the results of the three levels to make a comprehensive decision. From another perspective, the 1D convolutional aggregator has no restrictions on frame interval or sequence length, so our framework is robust to video length.

D. Training

Now we get frame-level, clip-level and video-level features. After fully connected layers, we get frame-based, clip-based and video-based predictions. During training, we optimize the following, a hierarchical loss function:

$$\mathcal{L} = \lambda_v * \mathcal{L}_{\text{video}} + \lambda_c * \mathcal{L}_{\text{clip}} + \lambda_f * \mathcal{L}_{\text{frame}} \tag{2}$$

Here $\mathcal{L}_{\text{video}}$, $\mathcal{L}_{\text{clip}}$, $\mathcal{L}_{\text{frame}}$ represent the binary classification loss of frame, clip and video respectively. λ_v, λ_c and λ_f are the respective weights. The loss for the binary classification is as follows:

$$\mathcal{L}_{\text{binary}} = -\frac{1}{N} \sum_{i=1}^{N} y_i \cdot \log\left(p\left(y_i\right)\right) + \left(1 - y_i\right) \cdot \log\left(1 - p\left(y_i\right)\right) \tag{3}$$

where N is the number of features ($N = 1$ for video-based, $N = 8$ for clip-based and $N = 16$ for frame-based), $p\left(y_i\right)$ is the probability of prediction as fake and y_i is the ground truth ($y_i = 0$ for real and $y_i = 1$ for fake).

Regardless of the dataset used, the EfficientNetB0 is firstly trained on the training split minimizing cross-entropy loss for binary classification to develop features to discern real faces from synthetics. The EfficientNetB0 is then extended with 1D MBConv aggregator and finally trained end-to-end.

IV. Experiments

In this section, we first describe the four datasets. Then, we provide details of the experimental settings to ensure reproducibility and end up by analyzing the reported results.

A. Datasets

We evaluated the proposed SDHF on four different datasets: DFDC [17], Celeb-DF [18], FaceForensics++ [9] and UADFV [7]. Table II shows the detailed descriptions of the datasets.

- **DFDC**: The Deepfake Detection Challenge (DFDC) [17] dataset contains a total of 119146 videos with face and audio manipulations. The video tampering scenes in this dataset are more complex and diverse. The provide training videos are divided into 50 numbered parts. We use the first 30 parts for training, parts from 30-39 for validation and the last 10 parts for testing.
- **Celeb-DF**: Celeb-DF [18] includes 590 original videos collected from YouTube with subjects of different ages, ethic groups and genders, and 5639 corresponding DeepFake videos. They reduce temporal flickering or synthetic faces in the DeepFake videos by incorporating temporal correlations among the detected face landmarks, so, these videos are visually smooth. We use the test videos divided by the authors for testing, and the remaining videos for training except the YouTube-real collection.
- **FaceForensics++**: FaceForensics++ [9] consists of 1000 original videos, which are manipulated by five different methods, with 720 in training and 140 for each validate

and test. Besides, the dataset covers three different versions based on compression including Raw, c23 and c40. In this experiment, we only use the Deepfakes subset of FaceForensics++ at c23 (FF++/DF).

- **UDAFV**: UADFV was collected from FakeApp to validate the algorithms in [7].

TABLE II
STATISTICS OF THE DATASETS

Datasets	Real		Fake		Balanced
	train	test	train	test	
DFDC	10437	4300	58619	20219	×
Celeb-DF	711	178	5299	340	×
FF++/DF	720	140	720	140	√
UADFV	-	49	-	49	√

B. Evaluation Metrics

The testing set of DFDC and Celeb-DF, including training set, is unbalanced, which means that if the model predicts most samples are fake, it can still get a higher accuracy. So we choose balanced accuracy [43] (bACC) as our evaluation metric, which is robust to unbalanced testing sets. In addition, we introduce log loss (log$\mathcal{L}$) to evaluate the confidence of the model, which drastically penalizes being both confident and wrong.

$$bACC = \frac{TPR + TNR}{2} \tag{4}$$

TABLE III
QUANTITATIVE RESULTS FOR EACH METHOD WITH INTRA-TEST

Methods	DFDC		Celeb-DF		FF++/DF	
	bACC↑	log$\mathcal{L}$ ↓	bACC↑	log$\mathcal{L}$ ↓	bACC↑	log$\mathcal{L}$ ↓
Conv-mean	0.8706	0.4106	0.9845	0.3366	0.9786	0.3405
Conv-LSTM	0.8731	0.3710	0.9860	0.0560	0.9786	0.1554
I3D	0.8217	0.4182	0.9743	0.0884	0.9714	0.1542
C3D	0.6954	0.4995	0.8862	0.5317	0.8964	0.4980
SDHF(Ours)	**0.9094**	**0.1986**	**0.9943**	**0.0400**	**0.9821**	**0.1172**

C. Baselines

We compared the SDHF with 4 other approaches.

- **Conv-mean**: The predictions in all frames are averaged to obtain video-based predictions.
- **Conv-LSTM** [14]: Using CNN To extract frame-level features, LSTM as an aggregator. There are three differences from [14] in our implementation: (1) we choose EfficientNet as our backbone. (2) instead of conduct face alignment, we maintain the original facial pose. (3) instead of small interval sampling, we choose discrete, large interval sampling. The video-level feature was averaged recurrent features across all time-steps.
- **C3D** [44]: A classic action recognition model that uses 3D convolution to simultaneously extract spatio-temporal information.
- **I3D** [45]: An efficient action recognition model extends 2D convolution to 3D convolution based on the InceptionV1 [46] network structure.

D. Implementation Details

Although other variants of EfficientNet may achieve higher performance, we still choose EfficientNetB0 as the backbone to verify the feasibility of our framework. For training frame-based EfficientNetB0, we initialize the model by pre-trained weights on ImageNet. The batch size is set to 64 and each sample is augmented with a probability of 0.1, including JPEG compress, gaussian noise, blur and gamma correction. In addition, all samples are flipped horizontally with a probability of 0.5. All models are trained using Adam optimizer with learning rate $\eta = 1e-4$, with anneals every 10000 steps by $\eta * 0.7$ until 30000 steps were reached.

For Conv-LSTM and SDHF, we use the previously trained EfficientNetB0 as the feature extractor. Then the EfficientNetB0 is extended with different aggregators. The batch size is set to 8 and each video takes 16 frames at equal intervals as a sample. All video-based models Conv-LSTM, SDHF, I3D and C3D, are trained using Adam optimizer with learning rate $\eta = 1e-5$, with anneals every 10000 steps by $\eta * 0.7$ until 30000 steps were reached. Note that the input size of C3D is 112x112, and the input size of other methods are 224x224. For simplicity, the different losses in SDHF are set to equal weights. In fact, depending on the size of the dataset, the optimal results may arrive earlier. Therefore, we save the model once every 2000 steps and select the one with the best validation result.

TABLE IV
BALANCED ACCURACY OF EACH METHOD WITH CROSS-TEST

Methods	Intra	Cross		
	DFDC	Celeb-DF	FF++/DF	UADFV
	bACC↑	bACC↑	bACC↑	bACC↑
Conv-mean	0.8706	0.7844	0.7929	0.7755
Conv-LSTM	0.8731	0.8000	0.8071	0.8163
I3D	0.8217	0.6663	0.7250	0.6837
C3D	0.6954	0.6713	0.6536	0.6939
SDHF(Ours)	**0.9094**	**0.8114**	**0.8464**	**0.8571**

E. Results and Analysis

Our experiments have two goals: compare our approach to previous works and investigate our approach's performance vs the baselines. The first is necessary to validate that SDHF can outperform other approaches. The latter allow us to understand the reason why SDHF performs well and provide guidance for future research directions.

Intra-test. In order to demonstrate the effectiveness of our approach, we first conducted intra-test, which means we independently train and test on DFDC, Celeb-DF and FF++/DF. Table III shows the balanced accuracy (bACC) and log loss (log$\mathcal{L}$) of all compared methods on three datasets. We observe that our SDHF achieves the best performance on all the datasets against other methods. It demonstrate that hierarchical framework is effective for DeepFake detection. For Celeb-DF and FF++/DF, Conv-mean can already achieve an accuracy of more than 97%, but it can only achieve an accuracy of 87.06% on DFDC, which shows that forgery scenarios of

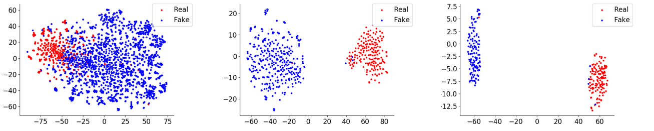
DFDC are more diverse and complex, and more challenging. There are multiple subjects, partial manipulations and person walking back and forth in DFDC, which is more complex and closer to the forged scenes of spreading videos on the social media. Note that although some works [47], [48] reported that Conv-LSTM does not even achieve an accuracy of 75% on DFDC, our results indicate that Conv-LSTM achieve better results than Conv-mean. The main difference between our experiments and others was that we select frames with a larger interval as a sample from a video, while other experiments select continuous frames with a smaller interval. The reason for processing the data in this way is that we argue multiple faces with small differences are almost the same after extracting features through CNNs. It is hard to capture temporal information on small interval faces and we only regard LSTM as an aggregator. This also prove that our sampling strategy is effective, which can sample more video information, such as multiple subjects, diverse facial expressions and head poses. From another perspective, Table III shows the performance of I3D and C3D is relatively inferior to Conv-mean. However, we argue that the reason for the poor experimental results is that the frame-by-frame face crop operation loses the continuity of data. If proper data processing is performed, better results can be obtained by the 3D convolution model.

Cross-test. Obviously, a simple EfficientNetB0 can achieve more than 85% balanced accuracy in discerning fake/real videos if training and testing videos are from the same source. In order to further prove the generalization ability of our SDHF, we conducted cross-test experiments, and all models were trained on DFDC, and tested on the other three datasets. As shown in Table IV, our approach achieves the highest balanced accuracy on all the datasets against other methods, which prove that our hierarchical detection framework can extract multi-level complementary features, including frame-level, clip-level and video-level, and has excellent generalization performance in the face of unknown datasets. To be noticed, our approach is superior to Conv-mean by 8.16% for UADFV, 5.35% for FF++/DF and 2.7% for Celeb-DF. The fake video in UADFV was collected from FakeApp, and there are more unforged frames and blurred frames in the video. Although this defect is easier for humans to identify, it can cause confusion for machines. So this results further demonstrate that our our approach can learn the ability to extract significant criteria from complex scenes, and can generalize this ability to other data.

Visualization. We adopted the video-level features obtained by SDHF for t-SNE [49] visualization. Visualization experiments also follow Intra-test, like Table III. Figure 6 shows that the video-level features learned by 1D MBConv aggregator can effectively distinguish fake from real videos.

F. Ablation Studies

In order to verify the effectiveness of our proposed hierarchical framework and loss function, we conducted a set of ablation experiments in cross-test. As shown in Table V, F, C and V represent frame-level, clip-level and video-level features, respectively. For example, V+C+F means that three levels of features are adopted simultaneously to make decisions. For fair comparison, we set equal weights for the three losses $\lambda_v = \lambda_c = \lambda_f$. From Table V and Table IV, when only video-level features is leveraged to make decisions, our approach superiors to Conv-mean by 1.51% balanced accuracy on DFDC with intra-test, 2.45% for Celeb-DF and 1.52% for FF++/DF with cross-test. When clip-level and frame-level features are added to jointly guide learning, our approach has a stable improvement in all datasets. The results indicate that there is a good complement of features at different levels in DeepFake detection.



(a) DFDC (b) Celeb-DF (c) FF++/DF

Fig. 6. t-SNE visualization of video-level features obtained by SDHF.

TABLE V
BALANCED ACCURACY OF SDHF WITH ABLATION STUDIES

Methods	Intra	Cross		
	DFDC	Celeb-DF	FF++/DF	UADFV
	bACC↑	bACC↑	bACC↑	bACC↑
V	0.8853	0.8089	0.8081	0.7755
V+F	0.8930	0.8059	0.8178	0.8163
V+C	0.8948	**0.8114**	0.8250	0.8265
V+F+C	**0.9094**	**0.8114**	**0.8464**	**0.8571**

V. CONCLUSION

In this work, we present a novel framework, named SDHF, for DeepFake detection. Our contribution lies in proposing a hierarchical framework, using 2D convolutional neural networks for frame-level features extraction and 1D MBConv aggregator to extract clip-level and video-level features, which can synthesize three different levels of features to make a comprehensive decision. Our approach is very easy to extend and can applied to any frame-based DeepFake detection model. In addition, our approach is robust to video length. The results of intra-test demonstrate that our approach achieves better performance compared to other methods. The results of cross-test prove that our hierarchical framework has excellent generalization performance in the face of unknown datasets. The results of ablation studies indicate that there is a good complement of features at different levels in DeepFake video detection.

REFERENCES

[1] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," *arXiv preprint arXiv:1312.6114*, 2013.

[2] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in neural information processing systems*, 2014, pp. 2672–2680.

[3] G. Antipov, M. Baccouche, and J. L. Dugelay, “Face aging with conditional generative adversarial networks,” in *2017 IEEE International Conference on Image Processing (ICIP)*, 2017.

[4] J. Thies, M. Zollhofer, M. Stamminger, C. Theobalt, and M. Niessner, “Face2face: Real-time face capture and reenactment of rgb videos,” *Communications of the ACM*, vol. 62, no. 1, pp. 96–104, 2019.

[5] K. Dale, K. Sunkavalli, M. K. Johnson, D. Vlasic, W. Matusik, and H. Pfister, “Video face replacement,” in *Proceedings of the 2011 SIGGRAPH Asia Conference*, 2011, pp. 1–10.

[6] Y. Li, M. Chang, and S. Lyu, “In ictu oculi: Exposing ai created fake videos by detecting eye blinking,” in *2018 IEEE International Workshop on Information Forensics and Security (WIFS)*, 2018, pp. 1–7.

[7] X. Yang, Y. Li, and S. Lyu, “Exposing deep fakes using inconsistent head poses,” in *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019, pp. 8261–8265.

[8] U. A. Ciftci and I. Demir, “Fakecatcher: Detection of synthetic portrait videos using biological signals,” *arXiv preprint arXiv:1901.02212*, 2019.

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

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

[11] L. M. Dang, S. I. Hassan, S. Im, and H. Moon, “Face image manipulation detection based on a convolutional neural network,” *Expert Systems with Applications*, vol. 129, pp. 156–168, 2019.

[12] J. Li, T. Shen, W. Zhang, H. Ren, D. Zeng, and T. Mei, “Zooming into face forensics: A pixel-level analysis,” *arXiv preprint arXiv:1912.05790*, 2019.

[13] J. Deng, J. Guo, Y. Zhou, J. Yu, I. Kotsia, and S. Zafeiriou, “Retinaface: Single-stage dense face localisation in the wild,” *arXiv preprint arXiv:1905.00641*, 2019.

[14] E. Sabir, J. Cheng, A. Jaiswal, W. AbdAlmageed, I. Masi, and P. Natarajan, “Recurrent convolutional strategies for face manipulation detection in videos,” *Interfaces (GUI)*, vol. 3, p. 1, 2019.

[15] 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*, 2019, pp. 0–0.

[16] M. Tan and Q. V. Le, “Efficientnet: Rethinking model scaling for convolutional neural networks,” *arXiv preprint arXiv:1905.11946*, 2019.

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

[18] Y. Li, P. Sun, H. Qi, and S. Lyu, “Celeb-DF: A Large-scale Challenging Dataset for DeepFake Forensics,” in *IEEE Conference on Computer Vision and Patten Recognition (CVPR)*, Seattle, WA, United States, 2020.

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

[20] Z. Xingjie, J. Song, and J.-I. Park, “The image blending method for face swapping,” in *2014 4th IEEE International Conference on Network Infrastructure and Digital Content*. IEEE, 2014, pp. 95–98.

[21] I. Korshunova, W. Shi, J. Dambre, and L. Theis, “Fast face-swap using convolutional neural networks,” in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 3677–3685.

[22] “Deepfakes.” [Online]. Available: https://github.com/deepfakes/faceswap

[23] Y. Nirkin, Y. Keller, and T. Hassner, “Fsgan: Subject agnostic face swapping and reenactment,” in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pp. 7184–7193.

[24] A. Pumarola, A. Agudo, A. M. Martinez, A. Sanfeliu, and F. Moreno-Noguer, “Ganimation: Anatomically-aware facial animation from a single image,” in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 818–833.

[25] L. Jiang, W. Wu, R. Li, C. Qian, and C. C. Loy, “Deeperforensics-1.0: A large-scale dataset for real-world face forgery detection,” *arXiv preprint arXiv:2001.03024*, 2020.

[26] P.-H. Huang, F.-E. Yang, and Y.-C. F. Wang, “Learning identity-invariant motion representations for cross-id face reenactment,” in *The IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[27] P. Zhou, X. Han, V. I. Morariu, and L. S. Davis, “Two-stream neural networks for tampered face detection,” in *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. IEEE, 2017, pp. 1831–1839.

[28] Y. Li and S. Lyu, “Exposing deepfake videos by detecting face warping artifacts,” *arXiv preprint arXiv:1811.00656*, 2018.

[29] Z. Guo, G. Yang, J. Chen, and X. Sun, “Fake face detection via adaptive residuals extraction network,” *arXiv preprint arXiv:2005.04945*, 2020.

[30] N. Bonettini, E. D. Cannas, S. Mandelli, L. Bondi, P. Bestagini, and S. Tubaro, “Video face manipulation detection through ensemble of cnns,” *arXiv preprint arXiv:2004.07676*, 2020.

[31] D. Güera and E. J. Delp, “Deepfake video detection using recurrent neural networks,” in *2018 15th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*. IEEE, 2018, pp. 1–6.

[32] P. Chen, T. Liang, J. Liu, J. Dai, and J. Han, “Forged facial video detection based on global temporal and local spatial feature,” *Journal of Cyber Security*, vol. 5, no. 2, pp. 73–83, 2020.

[33] X. Yang, Y. Li, and S. Lyu, “Exposing deep fakes using inconsistent head poses,” in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 8261–8265.

[34] L. Verdoliva, “Media forensics and deepfakes: an overview,” *arXiv preprint arXiv:2001.06564*, 2020.

[35] H. H. Nguyen, F. Fang, J. Yamagishi, and I. Echizen, “Multi-task learning for detecting and segmenting manipulated facial images and videos,” *arXiv preprint arXiv:1906.06876*, 2019.

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

[37] J. Hu, L. Shen, and G. Sun, “Squeeze-and-excitation networks,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 7132–7141.

[38] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, “Mobilenetv2: Inverted residuals and linear bottlenecks,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 4510–4520.

[39] M. Tan, B. Chen, R. Pang, V. Vasudevan, M. Sandler, A. Howard, and Q. V. Le, “Mnasnet: Platform-aware neural architecture search for mobile,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 2820–2828.

[40] L. Wan, M. Zeiler, S. Zhang, Y. Le Cun, and R. Fergus, “Regularization of neural networks using dropconnect,” in *International conference on machine learning*, 2013, pp. 1058–1066.

[41] K. He, X. Zhang, S. Ren, and J. Sun, “Deep Residual Learning for Image Recognition,” *arXiv e-prints*, p. arXiv:1512.03385, Dec. 2015.

[42] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, “Imagenet: A large-scale hierarchical image database,” in *2009 IEEE conference on computer vision and pattern recognition*. Ieee, 2009, pp. 248–255.

[43] K. H. Brodersen, C. S. Ong, K. E. Stephan, and J. M. Buhmann, “The balanced accuracy and its posterior distribution,” in *2010 20th International Conference on Pattern Recognition*, 2010, pp. 3121–3124.

[44] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri, “Learning spatiotemporal features with 3d convolutional networks,” in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 4489–4497.

[45] J. Carreira and A. Zisserman, “Quo vadis, action recognition? a new model and the kinetics dataset,” in *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 6299–6308.

[46] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, “Going deeper with convolutions,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 1–9.

[47] D. M. Montserrat, H. Hao, S. Yarlagadda, S. Baireddy, R. Shao, E. Bartusiak, J. Yang, D. Guera, F. Zhu, E. J. Delp *et al.*, “Deepfakes detection with automatic face weighting,” *arXiv preprint arXiv:2004.12027*, 2020.

[48] A. Kumar and A. Bhavsar, “Detecting Deepfakes with Metric Learning,” *arXiv e-prints*, p. arXiv:2003.08645, Mar. 2020.

[49] L. v. d. Maaten and G. Hinton, “Visualizing data using t-sne,” *Journal of machine learning research*, vol. 9, no. Nov, pp. 2579–2605, 2008.