Detecting Deepfake Videos with Temporal Dropout 3DCNN

Daichi Zhang[1,2*] , **Chenyu Li**[1,2*] , **Fanzhao Lin**[1,2] , **Dan Zeng**[3] and **Shiming Ge**[1,2†]
[1]Institute of Information Engineering, Chinese Academy of Sciences, Beijing 100095, China
[2]School of Cyber Security, University of Chinese Academy of Sciences, Beijing 100049, China
[3]School of Communication and Information Engineering, Shanghai University, Shanghai 200444, China
{zhangdaichi, lichenyu, linfanzhao, geshiming}@iie.ac.cn, dzeng@shu.edu.cn

Abstract

While the abuse of deepfake technology has brought about a serious impact on human society, the detection of deepfake videos is still very challenging due to their highly photorealistic synthesis on each frame. To address that, this paper aims to leverage the possible inconsistent cues among video frames and proposes a Temporal Dropout 3-Dimensional Convolutional Neural Network (TD-3DCNN) to detect deepfake videos. In the approach, the fixed-length frame volumes sampled from a video are fed into a 3-Dimensional Convolutional Neural Network (3DCNN) to extract features across different scales and identify whether they are real or fake. Especially, a temporal dropout operation is introduced to randomly sample frames in each batch. It serves as a simple yet effective data augmentation and can enhance the representation and generalization ability, avoiding model overfitting and improving detecting accuracy. In this way, the video-level classifier is trained to identify deepfake videos accurately and effectively. Extensive experiments on popular benchmarks clearly demonstrate the effectiveness and generalization capacity of our approach.

1 Introduction

Fake images and videos including facial information generated by digital manipulation, especially via deepfake methods, have become a great public concern recently [Suwajanakorn *et al.*, 2017], which has threatened politics and public social media area, such as personal revenge, evidence tampering, credibility harming, even political sabotage via falsifying media records, and public mood, existing legislation, and so on. Effective detection of those deepfake contents is an urging challenge around the world.

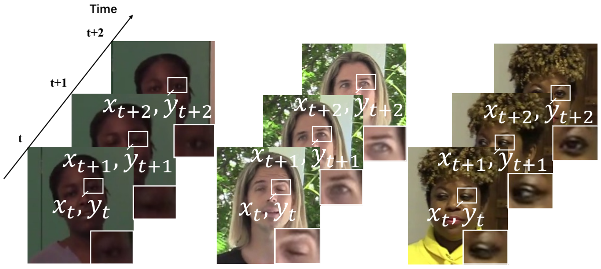

Figure 1: Motivation of this work. While deep generative methods nowadays can synthesize photo-realistic frames, temporal inconsistencies (for example, an unnatural blinking) within frames could serve as informative cues in deepfake video detection.

Existing methods usually train an image-level or video-level binary classifier using Deep Neural Networks(DNNs). Image-level methods aim to predict possibilities of each frame being fake [Wang *et al.*, 2017; Hsu *et al.*, 2018]. However, in this way, all the spatiotemporal information is discarded, which should have served as informative cues. In contrast, video-based methods, which take videos as input, are less explored. Architectures including Recurrent Neural Network (RNN), Long Short-Term Memory (LSTM) and 3DCNNs are commonly adopted [Guera and Delp, 2018]. Nevertheless, taking videos as input results in much higher complexity and difficulty in training.

To make full use of spatiotemporal information, this paper introduces an effective framework, Temporal Dropout 3-Dimensional Convolution Neural Network (TD-3DCNN). 3DCNNs recently have shown impressive performance on video representation and comprehending, which plays a key role in deepfake video detection. We introduce a 3D inception module, which learns to extract features from different scales. Temporal dropout, as the name implies, is performing dropout on the temporal level. In every epoch, a video within a mini-batch undergoes the following two operations: 1) we randomly sample a continuous sequence of a fixed length, named raw sequence; 2) then we randomly dropout part of the frames in the raw sequence, and the survived frames form the final sequence. In this way, the model is trained to leverage the spatiotemporal characteristics from incomplete samples and learn informative and robust representations.

The temporal dropout module contains double random sampling. Applying dropout to the segment amounts to further sampling a "shortened" video from it, which consists of all the frames that survive both fixed-length segmentation and

[*] First two authors contributed equally.
[†] Corresponding Author

dropout. A video with N frames, given the expected raw sequence length $C \leq N$ and input length n, can be therefore seen as a collection of $(N - C) \times \binom{C}{n}$ possible shortened videos. For each presentation of each training case, a new shortened video is sampled and used for training. In this way, the model learns to leverage the global spatiotemporal information without falling into traps of sampling biases.

Our main contributions are as follows. First, we propose an effective video-level deepfake detection framework, named TD-3DCNN. The 3DCNN employs 3D inception modules to leverage the spatiotemporal information of different scales and capture temporal inconsistencies between frames. Second, we propose a simple yet effective sampling mechanism, Temporal Dropout, which serves as an efficient data augmentation to improve the model's generalization and representation ability. Finally, we conduct extensive experiments on three benchmarks to show the effectiveness of our approach.

2 Related Work

2.1 Deepfake Video Generation and Detection

Deepfake, first proposed in 2017, refers to the technique of synthesizing fake media such as images and video using deep learning method [Tolosana *et al.*, 2020]. Generative adversarial networks (GANs) [Goodfellow *et al.*, 2014] have enabled a set of video manipulations including identity [Li *et al.*, 2020a], facial attributes and expressions [Thies *et al.*, 2016; Chu *et al.*, 2020], even body actions [Tulyakov *et al.*, 2018].

Currently, the detection methods for manipulated videos can be divided into image-based and video-based methods. The image-based ones focus on spatial artifacts, for example, the blending seams [Durall *et al.*, 2020]. In [Li *et al.*, 2020b] the authors predict whether an image is fake and its blending boundary at the same time. While such approaches can learn to detect spatial artifacts from a large amount of training data, they do not have good performance across different datasets.

To overcome such defects, some researchers hypothesized that the generated contents lack physiological signals which causes inconsistencies across frames, and proposed to take the temporal information in the video into consideration. The authors in [Ciftci *et al.*, 2020] use the pulse signal as the evidence for detection. In [Agarwal *et al.*, 2020], the authors refined the approach by detecting inconsistencies between the visemes and spoken phonemes. These methods may perform better than the image-based ones, but they are only applicable to particular kinds of attacks.

2.2 3DCNN for Deepfake Detection

Recently, 3DCNNs have shown impressive superiority on multiple tasks including motion recognition [Tran *et al.*, 2015], activity recognition [Donahue *et al.*, 2017], and human Re-ID [McLaughlin *et al.*, 2016]. This is because the 3DCNNs can better utilize the spatiotemporal information via 3D convolution and 3D pooling operations.

Inspired by the vanilla 3DCNN, a number of variants were proposed. I3D [Carreira and Zisserman, 2017] uses a bunch of RGB frames as input. It replaces 2D convolutional layers of the original Inception model with 3D convolutions for spatiotemporal modeling and inflates pre-trained weights of on ImageNet for initialization. Results showed that such inflation has the ability to improve 3D models. 3D ResNet [Hara *et al.*, 2018] and 3D ResNeXt are also inspired by I3D, extending initial 2D ResNet and 2D ResNeXt to spatiotemporal dimension for action recognition. Deviating from the original ResNet-bottleneck block, the ResNeXt-block introduces group convolutions, which divide the feature maps into small groups. In summary, 3DCNN has an impressive performance on video understanding and representation, which may also play a key role in deepfake video detection task.

3 The Proposed Approach

3.1 Problem Formulation

Given a video $V = \{f_i\}_{i=1}^{i=N}$ consisting of N frames, the objective of deepfake detection is to learn a binary classification model ϕ to tell fake from real. Image-level methods take single frame as input, performing frame-wise prediction:

$$\delta(V = true) = \cap_t \phi_i(f_t, \omega_{\phi_i}), \;\; t \in [1, N], \qquad (1)$$

where $\delta(V = true)$ is the predicted possibility of the video V being true, $\cap$ is the AND operation, ϕ_i denotes the image-level detector and ω_{ϕ_i} are its parameters. When these approaches treat each frame in the video as independent images, the spatiotemporal relationships between adjacent and nonadjacent frames are wasted.

Differently, video-level methods take a sequence consisting of multiple frames as input. To decrease the computation pressure of the model, usually a video sampling technique is adopted and the sampled volume is then sent into the detector:

$$\begin{aligned} \delta(V = true) &= \phi_v(F_t, \omega_{\phi_v}) \\ &= \phi_v(\{f_{t_1}, \ldots, f_{t_C}\}, \omega_{\phi_v}), \\ t_1, \ldots, t_C &\in [1, N] \end{aligned} \qquad (2)$$

where $F_t = \{f_{t_1}, f_{t_2}, ..., f_{t_C}\}$ is a video volume consisting of C frames, ϕ_v denotes the video-level detector and ω_{ϕ_v} are its parameters. Evidently, to learn a robust video-level detector ϕ_v, two main problems need to be addressed: 1) *how to sample representative frame sequence* F_t, and 2) *how to leverage the context and spatiotemporal information within and enhance the representation ability of the detector*.

In this paper, we propose to solve the two problems by introducing a simple yet effective sampling technique called Temporal Dropout (TD). In every epoch, a video within a mini-batch undergoes the following two operations: 1) we randomly sample a continuous sequence of a fixed length, named raw sequence; 2) then we randomly drop out part of the frames, and the survived frames form the final sequence. In this way, the model is trained to leverage the spatiotemporal information and perform accurate deepfake detection. The overall predicting process using TD can be formulated as:

$$\begin{aligned} \delta(V = true) &= \phi_v(F_t, \omega_{\phi_v}) \\ &= \phi_v(\{f_{t_1}, \ldots, f_{t_n}\}, \omega_{\phi_v}), \\ t_1 \leq t_2 \leq, \ldots, \leq t_n &\in [i, i + C) \; , \; i \in [0, N - C] \end{aligned} \qquad (3)$$

where $F_t \subset \{f_i, f_{i+1}, ..., f_{i+C-1}\}$ is obtained by randomly selecting n frames out of the raw sequence consisting of C continuous frames with a starting index i.

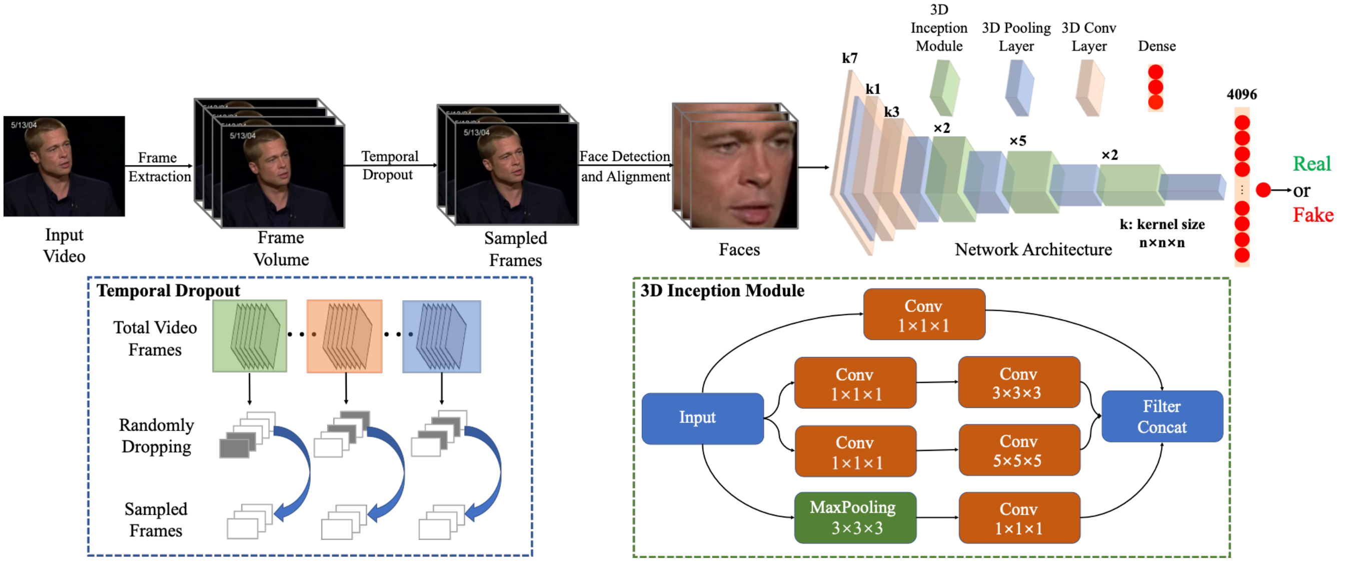

Figure 2: The framework of our proposed TD-3DCNN. The video frame volumes are sampled and augmented by the temporal dropout operation and fed into a 3DCNN, which enhances the feature representation ability and model generalizability, leading to an accurate and effective video-level deepfake detector.

3.2 TD-3DCNN

The framework of our proposed TD-3DCNN is shown in Fig. 2, consisting of a pre-processing stage and a feature extraction and classification stage. The pre-processing stage takes original videos as input and turns them into uniform format to facilitate the following process. The feature extraction and classification stage adopts a 3DCNN to learn robust representations via leveraging the context as well as spatiotemporal information. Given an input video, we first extract all frames. The complete frame volume is passed to the Temporal Dropout module and undergoes double random sampling operations. Then we extract the face area of each frame using a pre-trained face detection network. The faces are then resized, concatenated, and passed to the designed 3DCNN to tell whether the input video is fake or not.

The architecture of the 3DCNN is presented in the upper-right in Fig. 2. The first convolutional layer adopts the $7 \times 7 \times 7$ kernel, used to extract the low-level information. Then there are a pooling layer and a convolutional layer with $1 \times 1 \times 1$ kernel, to reduce the feature size and total parameters. The third convolutional layer using $3 \times 3 \times 3$ kernel is to extract the high-level information, followed by a pooling layer. Then there are three inception modules, each followed by a pooling layer, used to extract temporal features of different scales, enhancing the model's representation ability. Finally, the condensed feature is passed through a dense layer and a softmax layer to get the prediction result.

The receptive field is one of the primary factors that decide the representation ability. While 2-dimensional models treat the two spatial dimensions (horizontal and vertical) equally, it leaves ample freedom for 3-dimensional video processing models to inflate the operators along the temporal dimension. To extract and learn those features properly and enhance the video-level representation ability, we design a 3-dimensional Inception Module, detailed architecture of which is shown in lower-right in Fig. 2. The purpose of different kernel numbers and sizes in those four branches is to extract temporal features of different scales, capturing the videos' details and improving our classifier's performance. Worth noting, our proposed Inception Module use $3 \times 3 \times 3$ and $5 \times 5 \times 5$ kernel size in the second and third branch to extract different scale feature more effectively, which is different from the one in [Carreira and Zisserman, 2017].

3.3 Temporal Dropout

The algorithm of our TD module is described in Alg. 1. In each training epoch, after extracting all the frames from the training video, we first choose a random beginning index and sample a continuous frame sequence, named raw sequence. The length of the raw sequence length is $C = n \times \alpha$, where n is the desired length of model input and α is the amplification coefficient. Then we randomly drop $C - n$ frames out of the raw sequence, and obtain the final sequence at length n, which is then passed into 3DCNN for prediction.

By introducing our TD module, our 3DCNN can obtain different sequence frames as input in different epochs for the same video. It can not only extract the consistent local temporal information but also preserve the whole original sequence's global information. The dropout operation achieves a "Data Augmentation" effect. Both of those traits enhance the video-level representation ability of our model.

Two commonly used video sampling approaches, i.e., Systematic Sampling (SS) [Madow and Madow, 1944] and Continuous Sampling (CS) [Seppä, 2008], also work on the video level and are closely related to our Temporal Dropout (TD). SS samples at a fixed interval. It can well preserve the global temporal information but the obtained sequence itself is discrete, where the local temporal information is sacrificed.

Algorithm 1 Temporal Dropout

```
Require:
    $F$: original frames of input video;
    $N$: length of $F$;
    $n$: length of our TD-3DCNN input;
    $\alpha$: amplification coefficient;
Ensure:
    $F_t$: sampled video volume consisting of certain frames.

$index = random(1, N - n \times \alpha)$
$F_{raw} = F[index : index + n \times \alpha]$
$F_t = RandomChoose(F_{raw}; n \times \alpha, n)$
return $F_t$
```

Differently, CS samples a continuous sequence of a certain length, in the defined domain such as image or frequency domain. The obtained sequence in this way preserves local temporal consistency while losing global consistency. The difference with CS is that: after randomly sampling a continuous sequence, TD further enforces representation towards relationships between nonadjacent frames by dropping out part of the frames, leading to better global consistency.

In the training process, the continuous sampling preserves the local temporal information of original sequences and the randomly chosen beginning index in every iteration enforces preservation of the global information. The randomly temporal dropout operation acts as a data augmentation process: the classifier can always obtain different sequences. Thus we can efficiently enhance the representation ability of our model.

4 Experiments

4.1 Experimental Setting

We conduct experiments on three deepfake video datasets: the Celeb-DF(v2) [Li *et al.*, 2020c], DFDC [Dolhansky *et al.*, 2019] and FaceForensics++ [Rössler *et al.*, 2019]. Details of the three datasets and experimental settings are listed as follows. We implement all the models with PyTorch [Paszke *et al.*, 2019] on NVIDIA TITAN Xp.

The Celeb-DF(v2) dataset [Li *et al.*, 2020c] contains 590 original videos collected from YouTube with subjects of different ages, ethnic groups and genders, and 5,639 corresponding synthesized videos, which have similar visual quality on par with those circulated online. We follow the provided dividing principle to obtain the training and testing sets.

The Deepfake Detection Challenge (DFDC) dataset [Dolhansky *et al.*, 2019] consists of 19,197 real videos from 430 paid actors, and 100,000 fake videos, with accompanying labels describing whether they are deepfake videos. Fake videos are generated by facial manipulation techniques including DeepFakes, Face2Face and etc. DFDC considers different acquisition scenarios (i.e., indoors and outdoors), light conditions (i.e., day and night), distances from the person to the camera, and pose variations, among others. We divide our training, validation and testing sets by a ratio of 6:2:2. And finally, we obtain 7,528 videos as training set, 2,482 videos as validation set and 2,541 videos as testing set.

FaceForensics++ [Rössler *et al.*, 2019] is a dataset consisting of 1,000 original video sequences from Youtube, as well as corresponding manipulated videos using four different manipulation methods: Deepfakes[1], FaceSwap[2], NeuralTextures [Thies *et al.*, 2019] and Face2Face [Thies *et al.*, 2016]. To obtain the training, validation and testing sets, we randomly split the videos in the FaceForensics++ dataset in 6:2:2. Finally we obtain 4,074 videos as training set, 1,269 videos as validation set and 1,363 videos as testing set.

To prepare the data, we first use FFmpeg[3] to extract all the frames, then sample the frame sequence and extract the face area by a pre-trained MobileNet[4]. The faces are then resized into 224×224 images, which are the input. During training, we set the batch size as 16 and the total epoch is 50. The model is trained via Adam [Kingma and Ba, 2015] optimization with the global learning rate set as 10^{-5} and weight decay set as 10^{-6}. We adopt the cross-entropy as the loss function. The activation function of all layers is Relu function. All 3D convolution layers' stride are $1 \times 1 \times 1$ and all pooling layers' stride are $2 \times 2 \times 2$ using the Same padding. For the Temporal Dropout module, we set $n = 20, \alpha = 1.25$, which means that we sample continuous $20 \times 1.25 = 25$ frames and then randomly choose 20 frames from them.

4.2 Results

To validate the effectiveness of our TD-3DCNN framework, we perform comparisons with six state-of-the-art detectors on Celeb-DF(v2) and DFDC dataset: *Two-stream NN* [Zhou *et al.*, 2017], *MesoNet* [Afchar *et al.*, 2018], *Head Pose* [Yang *et al.*, 2019], *Visual Artifacts* [Matern *et al.*, 2019], *Multi-task* [Nguyen *et al.*, 2019], *Warping Artifacts* [Li and Lyu, 2019].

We use AUC score as our evaluation metric and Tab. 1 presents the results. Our method achieves the best performance on Celeb-DF(v2) and DFDC dataset, even 24.23% and 3.47% higher than the recent work [Li and Lyu, 2019], proving the proposed TD-3DCNN to be truly effective in deepfake video detection task. Worth noting, other methods' benchmark in DFDC used DFDC Preview instead of full DFDC dataset, which is more complicated and challenging. Obviously our TD-3DCNN has different performance on these different three datasets, which reflects the different complexity and difficulty of these two datasets.

Additionally, in order to test the stability and adaptability, we evaluate the TD-3DCNN on FaceForensics++ benchmark. Different from [Li and Lyu, 2019] where state-of-the-arts are evaluated only on a subset of FaceForensics++, we take the complete testing set containing all different classes for evaluation. Finally our TD-3DCNN presents a competitive performance of 72.2% AUC score. It suggests our model learns highly adaptive representations that allows stable detection over videos manipulated by varieties of methods.

To demonstrate the model's generalization ability, we also conduct experiments by training and testing across different datasets. Six metrics are adopted for evaluation: the accu-

[1] https://github.com/deepfakes/faceswap

[2] https://github.com/MarekKowalski/FaceSwap/

[3] https://ffmpeg.org/

[4] https://github.com/yeephycho/tensorflow-face-detection

Methods	Celeb-DF(v2)[Li *et al.*, 2020c]	DFDC[Dolhansky *et al.*, 2019]
Two-stream NN [Zhou *et al.*, 2017]	*53.80*	*61.40*
MesoNet [Afchar *et al.*, 2018]	*54.80*	*75.30*
Head Pose [Yang *et al.*, 2019]	*54.60*	*55.90*
Visual Artifacts [Matern *et al.*, 2019]	*55.10*	*66.20*
Multi-task [Nguyen *et al.*, 2019]	*54.30*	*53.60*
Warping Artifacts [Li and Lyu, 2019]	*64.60*	*75.50*
TD-3DCNN	**88.83**	**78.97**
Average Improvement	32.63	14.32

Table 1: AUC(%) comparisons of TD-3DCNN and other six state-of-the-art deepfake detection methods on Celeb-DF(v2) and DFDC datasets. Results in *italics* indicate that they were pulished in [Li *et al.*, 2020c], but not in the original work. In each column, the highest score is remarked in **bold**. The bottom row presents the the average improvement.

Train Set	Test Set	ACC(%)↑	AUC(%)↑	Logloss↓	Recall(%)↑	Precision(%)↑	F1(%)↑
Celeb-DF(v2)	Celeb-DF(v2)	**81.08**	**88.83**	**0.415**	**99.41**	68.98	82.42
	DFDC	74.85	60.75	0.532	85.44	**86.08**	**85.76**
	FaceForensics++	73.88	64.72	0.552	85.51	81.99	83.71
DFDC	Celeb-DF(v2)	66.60	64.20	0.636	**99.41**	66.40	79.62
	DFDC	**82.64**	**78.97**	**0.367**	98.77	**85.89**	**91.88**
	FaceForensics++	75.93	56.26	0.566	94.39	79.03	86.03
FaceForensics++	Celeb-DF(v2)	66.02	57.32	0.678	**100.0**	65.89	79.44
	DFDC	**82.21**	55.02	0.491	98.29	**83.36**	**90.20**
	FaceForensics++	79.09	**72.22**	**0.469**	99.63	79.14	88.21

Table 2: Experiments to demonstrate the generalization ability of TD-3DCNN. We train and cross evaluate the TD-3DCNN on three different datasets: Celeb-DF(v2), DFDC, FaceForensics++. We use following six metrics for evaluation: the accuracy (ACC), the area under the curve (AUC), Logloss, Recall, Precision and F1 score. In each column, the highest score is remarked in **bold**.

racy (ACC), the area under the curve (AUC), Recall, Precision, F1 score, and Logloss $\mathcal{L}_{log}$, formulated as follows:

$$\mathcal{L}_{log} = -\frac{1}{N}\sum_{i=1}^{N}[y_i \ln \hat{y}_i + (1-y_i)\ln(1-\hat{y}_i)], \quad (4)$$

where N is the size of training set. For the i th sample, y_i is the ground truth label and $\hat{y}_i$ is the prediction of the model.

As shown in Tab. 2, our TD-3DCNN models trained on different datasets have different performance when used to test other datasets. In most circumstances, metrics such as AUC and Logloss obtain the best when training and testing sets are the same. However some other metrics such as ACC, Recall, Precision and F1 are even better when using other different datasets as testing sets. For example, our model trained on FaceForensics++ gets higher ACC on DFDC (82.21%) and higher Recall on Celeb-DF (v2) (100%). Similar rules apply when model trained on a dataset is tested on different dataset, such as the model trained on Celeb-DF (v2) get a better F1 score (85.76%) on DFDC testing set and model trained on FaceForensics++ obtained better ACC (82.21%) on DFDC testing set. The reason for such results may be the different complexities of these three datasets, specifically, the videos in Celeb-DF (v2) are more complex and difficult to classify so the model trained on Celeb-DF (v2) obtains a better score than on the simpler datasets DFDC as well as FaceForensics++. And as we expected, Celeb-DF (v2) is the newest dataset with higher-quality compared to FaceForensics++ and DFDC. In some cases the performance on the test set is better than that on the training set, such as those trained on FaceForensics++ and tested on Celeb-DF (v2) or DFDC. We suspect it is because the video quality in different datasets varies a lot. These results reflect that our TD-3DCNN has a good generalization ability when facing deepfake video detection tasks. It can well leverage the inconsistent cues among video frames and obtains a strong representation of input video.

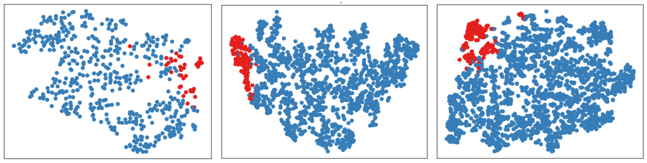

Figure 3: Visualization of representations on Celeb-DF(v2), FaceForensics++ and DFDC (from left to right). Red: Real, Blue: Fake.

4.3 Discussions

Discussion on Representation Learning We visualize the representations learned with our TD-3DCNN using t-SNE [Laurens and Hinton, 2008] on the test sets of three benchmarks. As shown in Fig. 3, the representations can effectively cluster real and fake videos.

	Test Train	SS	CS	TD	Avg
ACC(%)↑	SS	70.85	70.08	70.85	70.59
	CS	70.46	74.52	72.59	72.52
	TD	**76.06**	**81.08**	**80.70**	**79.28**
AUC(%)↑	SS	73.12	74.13	75.05	74.10
	CS	82.24	87.23	85.89	85.12
	TD	**85.02**	**88.83**	**88.36**	**87.40**
$\mathcal{L}_{log}\downarrow$	SS	0.755	0.741	0.788	0.764
	CS	0.770	0.582	0.613	0.655
	TD	**0.561**	**0.415**	**0.430**	**0.469**

Table 3: The results of 3DCNN adopting different sampling methods for training and testing on Celeb-DF(v2) dataset. ↑ means that for this metric, a bigger value is preferred and vise versa. For each metric-test pair(sub-column), the highest score is remarked in **bold**.

Discussion on Sampling Methods To delve into the effectiveness of the temporal dropout operation, we adopt two widely used sampling methods, SS and CS, along with TD for training and testing, separately, and use ACC, AUC and Logloss for evaluation. The results on Celeb-DF(v2) are shown in Tab. 3. From the table, several conclusions can be drawn. First, during testing, CS usually performs the best. It suggests that in the process of deepfake video detection, the spatiotemporal information between adjacent frames plays an essential part. Sequences sampled by SS and TD both suffer from temporal discontinuity, while CS could preserve the local continuity of the sampled sequences, leading to an evident performance gain. Second, models trained using TD always achieve the highest ACC and AUC, and the lowest $\mathcal{L}_{log}$, no matter what sampling method is adopted at testing time. It suggests that TD could effectively benefit the representation learning. Third, at testing time, TD could achieve comparable performance with CS, especially for models trained with TD. It indicates that TD not only helps the model learn the global representation, but also preserves most of the spatiotemporal information within the local segment. In general, TD is a simple yet effective technique that helps leverage the spatiotemporal information in both global and local perspective.

Discussion on Consistency To further demonstrate the representation ability, during testing time, we sample multiple times for the test video and calculated the average of metrics. We introduce M_{con} to describe the consistency of multiple predictions, formulated as Eq. 5. Higher M_{con} means higher stability, and therefore, suggest that the model learns a better representation over the complete video.

$$M_{con} = \frac{1}{S \times m} \sum_{i=1}^{S} \sum_{j=1}^{m} [y_i P_{ij} + (1 - y_i)(m - P_{ij})], \quad (5)$$

where S is the size of training set, m is the repeated times of sampling, y_i is the ground truth label, and P_i is the frequency of prediction being fake for the j th sample of the i th video. As the results in Tab. 4 shown, our TD-3DCNN's performance slightly fluctuates with m increasing and stabilizes eventually. It performs the best on all metrics for all m values. In comparison, the other two methods, SS especially, the ACC and $\mathcal{L}_{log}$ metric endure more violent fluctuations as m goes up. These results imply that our TD-3DCNN can get more stable and robust predictions when given different parts of fake videos. And this reliability comes from the model's strong video-level representation, which contain rich spatiotemporal information in the original video.

		SS	CS	TD
ACC(%)↑	m=1	70.08	74.52	**81.08**
	m=3	73.94	72.97	**80.88**
	m=5	75.87	73.94	**82.43**
	m=7	73.55	72.97	**81.08**
	m=9	74.32	73.55	**81.85**
$\mathcal{L}_{log}\downarrow$	m=1	0.747	0.582	**0.415**
	m=3	0.613	0.581	**0.402**
	m=5	0.556	0.578	**0.392**
	m=7	0.584	0.573	**0.394**
	m=9	0.551	0.566	**0.396**
$M_{con}\uparrow$	m=1	0.733	0.734	**0.793**
	m=3	0.703	0.744	**0.805**
	m=5	0.717	0.740	**0.814**
	m=7	0.708	0.738	**0.803**
	m=9	0.722	0.744	**0.804**

Table 4: Average results on Celeb-DF(v2) of models with different sampling methods over m times of CS for the same test video. Here, ACC denotes accuracy, $\mathcal{L}_{log}$ and M_{con} are defined as Eq. 4, 5, separately. In each row, the highest score is remarked in **bold**.

5 Conclusion

In this paper, we proposed a Temporal Dropout 3-dimensional Convolutional Neural Network (TD-3DCNN) to detect deepfake videos. We first design a 3DCNN architecture with an introduced 3D Inception Module, which extracts features of the video on different scales. Then we introduce a Temporal Dropout (TD) module to leverage the inconsistent cues among video frames, improving our model's representation and generalization ability. Finally, we extensively evaluated the proposed model on public deepfake datasets, and our TD-3DCNN exhibits an impressive performance on detecting and generalization ability, surpassing the state of the arts. In the future, we will integrate other representation learning methods in our deepfake detection framework and extend the temporal dropout idea to more video understanding applications.

Acknowledgements

This work was partially supported by grants from the National Key Research and Development Plan (2020AAA0140001), National Natural Science Foundation of China (61772513), Beijing Natural Science Foundation (L192040) and the project from Beijing Municipal Science and Technology Commission (Z191100007119002). Shiming Ge is also supported by the Youth Innovation Promotion Association, Chinese Academy of Sciences.

References

[Afchar *et al.*, 2018] Darius Afchar, Vincent Nozick, Junichi Yamagishi, and Isao Echizen. Mesonet: a compact facial video forgery detection network. In *WIFS*, pages 1–7, 2018.

[Agarwal *et al.*, 2020] Shruti Agarwal, Hany Farid, Ohad Fried, and Maneesh Agrawala. Detecting deep-fake videos from phoneme-viseme mismatches. In *CVPRW*, pages 660–661, 2020.

[Carreira and Zisserman, 2017] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *CVPR*, pages 6299–6308, 2017.

[Chu *et al.*, 2020] Mengyu Chu, You Xie, Jonas Mayer, Laura Leal-Taixé, and Nils Thuerey. Learning temporal coherence via self-supervision for gan-based video generation. *ACM TOG*, 39(4):75:1–13, 2020.

[Ciftci *et al.*, 2020] Umur Aybars Ciftci, Ilke Demir, and Lijun Yin. How do the hearts of deep fakes beat? deep fake source detection via interpreting residuals with biological signals. In *IJCB*, pages 1–10, 2020.

[Dolhansky *et al.*, 2019] Brian Dolhansky, Russ Howes, Ben Pflaum, Nicole Baram, and Cristian Canton-Ferrer. The deepfake detection challenge (DFDC) preview dataset. *CoRR*, abs/1910.08854, 2019.

[Donahue *et al.*, 2017] Jeff Donahue, Lisa Anne Hendricks, Marcus Rohrbach, Subhashini Venugopalan, Sergio Guadarrama, Kate Saenko, and Trevor Darrell. Long-term recurrent convolutional networks for visual recognition and description. *IEEE TPAMI*, 39(4):677–691, 2017.

[Durall *et al.*, 2020] Ricard Durall, Margret Keuper, and Janis Keuper. Watch your up-convolution: Cnn based generative deep neural networks are failing to reproduce spectral distributions. In *CVPR*, pages 7890–7899, 2020.

[Goodfellow *et al.*, 2014] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron C. Courville, and Yoshua Bengio. Generative adversarial nets. In *NeurIPS*, pages 2672–2680, 2014.

[Guera and Delp, 2018] David Guera and Edward J. Delp. Deepfake video detection using recurrent neural networks. In *AVSS*, pages 1–6, 2018.

[Hara *et al.*, 2018] Kensho Hara, Hirokatsu Kataoka, and Yutaka Satoh. Can spatiotemporal 3d cnns retrace the history of 2d cnns and imagenet? In *CVPR*, pages 6546–6555, 2018.

[Hsu *et al.*, 2018] Chih-Chung Hsu, Chia-Yen Lee, and Yi-Xiu Zhuang. Learning to detect fake face images in the wild. In *IS3C*, pages 388–391, 2018.

[Kingma and Ba, 2015] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015.

[Laurens and Hinton, 2008] Van Der Maaten Laurens and Geoffrey Hinton. Visualizing data using t-sne. *JMLR*, 9(2605):2579–2605, 2008.

[Li and Lyu, 2019] Yuezun Li and Siwei Lyu. Exposing deepfake videos by detecting face warping artifacts. In *CVPRW*, pages 46–52, 2019.

[Li *et al.*, 2020a] Lingzhi Li, Jianmin Bao, Hao Yang, Dong Chen, and Fang Wen. Advancing high fidelity identity swapping for forgery detection. In *CVPR*, pages 5074–5083, 2020.

[Li *et al.*, 2020b] Lingzhi Li, Jianmin Bao, Ting Zhang, Hao Yang, Dong Chen, Fang Wen, and Baining Guo. Face x-ray for more general face forgery detection. In *CVPR*, pages 5001–5010, 2020.

[Li *et al.*, 2020c] Yuezun Li, Xin Yang, Pu Sun, Honggang Qi, and Siwei Lyu. Celeb-df: A large-scale challenging dataset for deepfake forensics. In *CVPR*, pages 3204–3213, 2020.

[Madow and Madow, 1944] William G. Madow and Lillian H. Madow. On the theory of systematic sampling, i. *The Annals of Mathematical Statistics*, 15(1):1–24, 1944.

[Matern *et al.*, 2019] F. Matern, C. Riess, and M. Stamminger. Exploiting visual artifacts to expose deepfakes and face manipulations. In *WACVW*, pages 83–92, 2019.

[McLaughlin *et al.*, 2016] Niall McLaughlin, Jesús Martínez del Rincón, and Paul C. Miller. Recurrent convolutional network for video-based person re-identification. In *CVPR*, pages 1325–1334, 2016.

[Nguyen *et al.*, 2019] Huy H Nguyen, Fuming Fang, Junichi Yamagishi, and Isao Echizen. Multi-task learning for detecting and segmenting manipulated facial images and videos. In *BTAS*, pages 1–8, 2019.

[Paszke *et al.*, 2019] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. In *NeurIPS*, pages 8026–8037, 2019.

[Rössler *et al.*, 2019] Andreas Rössler, Davide Cozzolino, Luisa Verdoliva, Christian Riess, Justus Thies, and Matthias Nießner. Faceforensics++: Learning to detect manipulated facial images. In *ICCV*, pages 1–11, 2019.

[Seppä, 2008] Mika Seppä. Continuous sampling in mutual-information registration. *IEEE TIP*, 17(5):823–826, 2008.

[Suwajanakorn *et al.*, 2017] Supasorn Suwajanakorn, Steven M. Seitz, and Ira Kemelmacher-Shlizerman. Synthesizing obama: learning lip sync from audio. *ACM TOG*, 36(4):95:1–13, 2017.

[Thies *et al.*, 2016] Justus Thies, Michael Zollhofer, Marc Stamminger, Christian Theobalt, and Matthias Nießner. Face2face: Real-time face capture and reenactment of rgb videos. In *CVPR*, pages 2387–2395, 2016.

[Thies *et al.*, 2019] Justus Thies, Michael Zollhöfer, and Matthias Nießner. Deferred neural rendering: image synthesis using neural textures. *ACM TOG*, 38(4):66:1–12, 2019.

[Tolosana *et al.*, 2020] Rubén Tolosana, Sergio Romero-Tapiador, Julian Fiérrez, and Rubén Vera-Rodríguez. Deepfakes evolution: Analysis of facial regions and fake detection performance. In *ICPR Workshops(5)*, pages 442–456, 2020.

[Tran *et al.*, 2015] Du Tran, Lubomir D. Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. Learning spatiotemporal features with 3d convolutional networks. In *ICCV*, pages 4489–4497, 2015.

[Tulyakov *et al.*, 2018] Sergey Tulyakov, Ming-Yu Liu, Xiaodong Yang, and Jan Kautz. Mocogan: Decomposing motion and content for video generation. In *CVPR*, pages 1526–1535, 2018.

[Wang *et al.*, 2017] Yu Wang, Luca Bondi, Paolo Bestagini, Stefano Tubaro, David J Edward Delp, et al. A counter-forensic method for cnn-based camera model identification. In *CVPRW*, pages 28–35, 2017.

[Yang *et al.*, 2019] Xin Yang, Yuezun Li, and Siwei Lyu. Exposing deep fakes using inconsistent head poses. In *ICASSP*, pages 8261–8265, 2019.

[Zhou *et al.*, 2017] Peng Zhou, Xintong Han, Vlad I. Morariu, and Larry S. Davis. Two-stream neural networks for tampered face detection. In *CVPRW*, pages 1831–1839, 2017.