Learning Self-Consistency for DeepFake Detection
Supplementary Material

1. Video-Level Comprehensive Results

Due to the space limit, we mainly reported the performance of our models in term of AUC in the main text. Here we provide more results in terms of AUC, AP, and EER for Table 4 and 5 in the main text. Note that these results are at *video-level*, computed by averaging the classification scores of all video frames. The experiments are conducted under the *cross-dataset* setting, in which we train our model only with real data from the raw version of FF++ [15] and the fake/positive samples are generated by I2G (more detailed are included in Sec. 4.2 in the main text).

Method	Test Set	Evaluation Metrics (%)		
		AUC	AP	EER
PCL + I2G	DF [5]	100.00	100.00	0.00
	F2F [17]	98.97	99.32	3.57
	FS [8]	99.86	99.86	1.43
	NT [18]	97.63	98.20	6.43
	FF++ [15]	99.11	99.80	3.57
	DFD [4]	99.07	99.89	4.42
	DFR [7]	99.41	99.51	3.48
	CD1 [10]	98.30	98.97	7.89
	CD2 [10]	90.03	94.45	17.98
	DFDC [2]	67.52	69.99	37.18
	DFDC-P [3]	74.37	82.94	31.87

Table 1: *Comprehensive evaluation of our model in terms of video-level AUC, AP, and EER on seven datasets.*

Additional qualitative results are shown in Fig. 1. These images are randomly chosen from CD2 [10] and DFDC [2] test sets, which are currently the most challenging datasets in deepfake detection. The visualization is obtained by up-sampling the 2D global heatmap $\widehat{\mathcal{M}}$ to the size of $H \times W$ to match the input size, where $\widehat{\mathcal{M}}$ is generated by fusing the predicted 4D consistency volume $\widehat{\mathbf{V}}$.

2. Frame-Level Results on Celeb-DF-v2

Deepfake detection accuracy is usually reported at the video-level. Methods aggregate the frame-level scores to form video-level predictions using various strategies, *e.g.*, averaging (ours), confident strategy [16], LSTM [6]. Here we compare our model with other state-of-the-art methods in terms of frame-level AUC on CD2 [10]. All models are trained under the *cross-dataset* setting (see Sec. 4.2 in the main text for more details). This means they are trained on FF++ and evaluated on CD2. As shown in Table 2, our model outperforms other state-of-the-art method [11] by over 8%. Note the baseline results are directly cited from Masi *et al.* [11]. Different compression levels of FF++ are adopted by the methods, *e.g.*, c23 for Zhao *et al.* [20], c40 for Masi *et al.* [11], and raw for ours.

Method	CD2 (Frame-Level AUC (%))
Two-stream [21]	53.8
Meso4 [1]	54.8
MesoInception4	53.6
HeadPose [19]	54.6
FWA [9]	56.9
VA-MLP [12]	55.0
VA-LogReg	55.1
Xception-raw [15]	48.2
Xception-c23	65.3
Xception-c40	65.5
Multi-task [13]	54.3
Capsule [14]	57.5
DSP-FWA [9]	64.6
Zhao *et al.* [20]	67.4
Masi *et al.* [11]	73.4
PCL + I2G	**81.8**

Table 2: *Cross-dataset evaluation of our model in terms of frame-level AUC on CD2 dataset.* The performances of existing methods are cited for comparison.

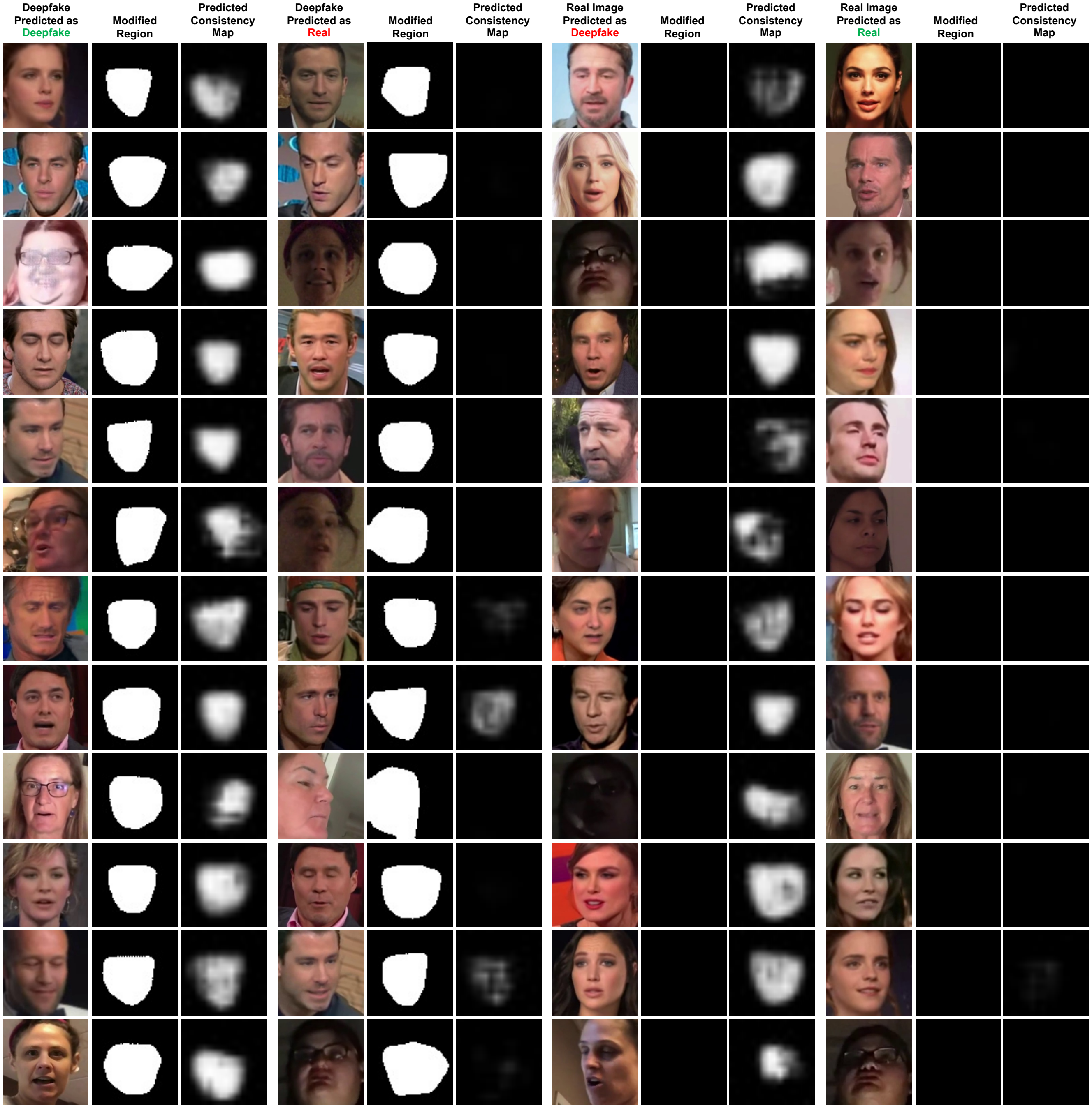
Deepfake Predicted as Deepfake
Modified Region
Predicted Consistency Map
Deepfake Predicted as Real
Modified Region
Predicted Consistency Map
Real Image Predicted as Deepfake
Modified Region
Predicted Consistency Map
Real Image Predicted as Real
Modified Region
Predicted Consistency Map

Figure 1: *Visualization of the predicted consistency maps $\widehat{\mathcal{M}}$, which try to localize the modified regions.* We use the model trained with real videos of FF++ augmented by I2G in the cross-dataset, and the predictions are computed from the predicted consistency volume, as mentioned in Section 3.1 in the main text. The ground truth modified regions are generated by DSSIM, as discussed in Section 4.2 in the main text.

3. Different Backbones

In this paper, we adopt ResNets as backbone, as they are among the most popular classification networks; an example of the PCL architecture is illustrated in Fig. 2. Our contribution does not rely on any particular choice of the model architecture. However, it is still interesting to discover if increasing the model capacity improves the cross-dataset generalization. We build our model with ResNets of various depths, and report the results in Table 3. The performance increase is noticeable from ResNet-18 to ResNet-50 but diminishes as we go deeper.

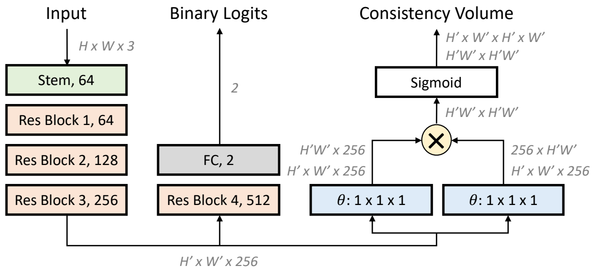

Figure 2: *An example of PCL architecture,* where ResNet-34 is adopted as backbone. The features are shown as the shape of their tensors, and proper reshaping is performed. $\otimes$ denotes matrix multiplication, and θ is 1×1×1 convolution.

Backbone	Test Set (AUC (%))				Avg
	DFR	CD2	DFDC	DFDC-P	
ResNet-18	96.92	79.59	58.22	68.23	75.74
ResNet-34	99.41	90.03	67.52	74.37	82.83
ResNet-50	99.13	90.70	**70.69**	**75.10**	**83.90**
ResNet-152	**99.5**	**90.88**	70.42	69.77	82.64

Table 3: *Ablation study of different backbones.* The models are all trained with $\lambda = 10$. The performance saturates as the depth of the model increases.

4. Test Set Statistics

Test Set	# Real / Fake Videos
FaceForensics++ (FF++) [15]	140 / 560
FF++ - Deepfakes (DF) [5]	140 / 140
FF++ - Face2Face (F2F) [17]	140 / 140
FF++ - FaceSwap (FS) [8]	140 / 140
FF++ - NeuralTextures (NT) [18]	140 / 140
DeepfakeDetection (DFD) [4]	363 / 3431
Celeb-DF-v1 (CD1) [10]	38 / 62
Celeb-DF-v2 (CD2) [10]	178 / 340
DFDC Public (DFDC) [2]	2000 / 2000
DFDC Preview (DFDC-P) [3]	276 / 504
DeeperForensics-1.0 (DFR) [7]	201 / 201

Table 4: *Statistics of real and fake videos in the test sets.*

5. Computational Complexity

Each vector of size C in the source feature map of size $H/P{\times}W/P{\times}C$ corresponds to a $P{\times}P$ patch in the input image, and P is the down-sampling factor. The additional computations from PCL consist of (1) embedding feature map into size $H/P{\times}W/P{\times}C'$ with $O(CC'HW/P^2)$ flops and (2) computing pair-wise consistency with $O(C'H^2W^2/P^4)$ flops. In practice, we use H=W=256, C=256, C'=128, P=16, and ResNet-34 as backbone, and PCL contributes 48.12M FLOPs, 65.5K parameters and 0.0009 seconds to a total of 9.62G FLOPs, 21.3M parameters and 0.0234 seconds, for a forward pass of a single image, running on one NVIDIA Tesla V100 GPU with 16GB of memory.

6. Consistency Map Generation

For visualization, we assume there are two source feature groups in each image. The top-left image patch belongs to group 0 and dissimilar patches to it belong to group 1. For a patch at (h, w) with a $H{\times}W$ consistency score matrix $\widehat{M}^{\mathcal{P}_{h,w}} = \{\widehat{m}_{i,j}^{\mathcal{P}_{h,w}}\}$, we get a soft group assignment as its cosine similarity to the top-left patch, $\widehat{m}_{0,0}^{\mathcal{P}_{h,w}}$. We calculate the corresponding grayscale value of this patch in the consistency map by averaging $|\widehat{m}_{0,0}^{\mathcal{P}_{h,w}} - \widehat{m}_{i,j}^{\mathcal{P}_{h,w}}|$. In this way, similar grayscale values in the visualization indicate the corresponding patches are likely to belong to the same source feature group.

References

[1] Darius Afchar, Vincent Nozick, Junichi Yamagishi, and Isao Echizen. Mesonet: a compact facial video forgery detection network. In *Proc. Workshop on Information Forensics and Security*, 2018.

[2] Brian Dolhansky, Joanna Bitton, Ben Pflaum, Jikuo Lu, Russ Howes, Menglin Wang, and Cristian Canton Ferrer. The deepfake detection challenge dataset. *arXiv:2006.07397*, 2020.

[3] Brian Dolhansky, Russ Howes, Ben Pflaum, Nicole Baram, and Cristian Canton Ferrer. The deepfake detection challenge (dfdc) preview dataset. *arXiv:1910.08854*, 2019.

[4] Nick Dufour and Andrew Gully. Contributing data to deepfake detection research, 2019. https://ai.googleblog.com/2019/09/contributing-data-to-deepfake-detection.html.

[5] FaceSwapDevs. Deepfakes, 2019. https://github.com/deepfakes/faceswap.

[6] David Güera and Edward J Delp. Deepfake video detection using recurrent neural networks. In *IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, 2018.

[7] Liming Jiang, Ren Li, Wayne Wu, Chen Qian, and Chen Change Loy. DeeperForensics-1.0: A large-scale dataset for real-world face forgery detection. In *Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2886–2895, Jun. 14–19 2020.

[8] Marek Kowalski. FaceSwap, 2018. https://github.com/MarekKowalski/FaceSwap.

[9] Yuezun Li and Siwei Lyu. Exposing deepfake videos by detecting face warping artifacts. In *Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, Jun. 16–20 2019.

[10] Yuezun Li, Xin Yang, Pu Sun, Honggang Qi, and Siwei Lyu. Celeb-DF: A large-scale challenging dataset for deepfake forensics. In *Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3207–3216, Jun. 14–19 2020.

[11] Iacopo Masi, Aditya Killekar, Royston Marian Mascarenhas, Shenoy Pratik Gurudatt, and Wael AbdAlmageed. Two-branch recurrent network for isolating deepfakes in videos. In *Proc. European Conference on Computer Vision*, Aug. 23–28 2020.

[12] Falko Matern, Christian Riess, and Marc Stamminger. Exploiting visual artifacts to expose deepfakes and face manipulations. In *Proc. Winter Applications of Computer Vision Workshops*, 2019.

[13] Huy H. Nguyen, Fuming Fang, Junichi Yamagishi, and Isao Echizen. Multi-task learning for detecting and segmenting manipulated facial images and videos. In *Proc. Biometrics Theory, Applications and Systems*, Sep. 23–26 2019.

[14] Huy H Nguyen, Junichi Yamagishi, and Isao Echizen. Capsule-forensics: Using capsule networks to detect forged images and videos. In *Proc. International Conference on Acoustics, Speech and Signal Processing*, 2019.

[15] Andreas Rössler, Davide Cozzolino, Luisa Verdoliva, Christian Riess, Justus Thies, and Matthias Nießner. FaceForensics++: Learning to detect manipulated facial images. In *Proc. IEEE/CVF International Conference on Computer Vision*, pages 1–11, Oct. 27–Nov. 2 2019.

[16] Selim Seferbekov. dfdc winning solution, 2020. https://github.com/selimsef/dfdc_deepfake_challenge#averaging-predictions.

[17] J. Thies, M. Zollhöfer, M. Stamminger, C. Theobalt, and M. Nießner. Face2Face: Real-time Face Capture and Reenactment of RGB Videos. In *Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2387–2395, Jun. 26–Jul. 1 2016.

[18] Justus Thies, Michael Zollhöfer, and Matthias Nießner. Deferred neural rendering: Image synthesis using neural textures. In *ACM Transactions on Graphics*, volume 38, pages 1–12, 2019.

[19] Xin Yang, Yuezun Li, and Siwei Lyu. Exposing deep fakes using inconsistent head poses. In *International Conference on Acoustics, Speech, and Signal Processing*, 2018.

[20] Hanqing Zhao, Wenbo Zhou, Dongdong Chen, Tianyi Wei, Weiming Zhang, and Nenghai Yu. Multi-attentional deepfake detection. In *Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021.

[21] P. Zhou, X. Han, V. I. Morariu, and L. S. Davis. Two-stream neural networks for tampered face detection. In *Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, Jun. 21–26 2017.