Supplementary Material for TI^{2}Net: Temporal Identity Inconsistency Network for Deepfake Detection

Baoping Liu[1], Bo Liu[1], Ming Ding[2], Tianqing Zhu[1], Xin Yu[1]
[1]University of Technology Sydney, NSW, Australia
[2] Data61, CSIRO, Sydney, NSW, Australia
baoping.liu@student.uts.edu.au, ming.ding@data61.csiro.au
{Bo.liu, Tianqing.Zhu, Xin.Yu}@uts.edu.au

This supplementary provides additional details, results and analysis of the method proposed in the main paper. In Section 1, we provide details of model training. In Section 2, we compare our methods with some close-set baselines. In Section 3, we show some details and samples of image degradation settings.

1. Implementation Details of Experiments

Pre-processing: To compress the cost of hard drive I/O during training and improve training efficiency, we first calculate the identity vectors of all frames in all videos and save the identity vectors as files. During the online training stage, the dataset loader simply reads the saved identity vector files and generates the real sequence set and the fake sequence set.

Training details: For the RNN, we adopt the bidirectional Gated Recurrent Unit (GRU) with the number of RNN units as 512. Following the RNN are two fully-connected layers as the classification head. To avoid the model over-fitting, dropout mechanism with a dropout rate of 0.2 and 0.5 is applied to the RNN input and each layer of the classification head, respectively. The hyperparameter λ_1 are λ_2 set as 1 and 0.1, respectively. The triplet loss margin α is set as 1. We use the SGD Adam optimizer with an initial learning rate of lr=0.0005. We train 100 epochs until the loss does not drop significantly.

After generating the real sequence set and the fake sequence set, we adopt an 8:2 data split, i.e., 80% for training and 20% for testing. For datasets adopted for testing, we adopt the full dataset of CelebDFv1 and CelebDFv2. As for DFD and Deeper datasets, known as very large datasets, we adopt a subset of them and generate over 20,000 sequences. The training and test experiments are conducted with a GTX 3090 Graphics card with a graphics memory of 24G.

Table 1: Generalization ability evaluation in terms of video-level AUC (%) on different testing datasets.

Methods	Testing Datasets			
	DFD	Deeper	CDF1	CDF2
A&B [1]	77.65	82.33	86.52	88.2
ICT [2]	**93.17**	**99.25**	**96.41**	**94.43**
TI^2Net(Ours)	**72.03**	**76.08**	**66.65**	**68.22**

2. Comparison with Close-set Baselines

As open-set baselines and close-set baselines have very different training and inference manners. Close-set detectors usually recognize fake videos by comparing the matching with the identities in the reference set. Therefore, close-set detectors usually achieve higher classification performance. In detail, we compare our methods with two close-set baselines:

(1) **A&B** [1]: A Deepfake detector that integrates behaviour and appearance of identities.

(2) **ICT** [2]: A Deepfake detector based on spatial identity inconsistency.

The A&B baseline is trained on the FF++ dataset and tested on all these datasets, the reference set is constructed by mixing the video clips of all identities in these datasets. The results of ICT are from the report of the paper. Since close-set baselines have reference sets, the comparison of our methods to close-set baselines is unfair for our methods in terms of training manner and datasets, it's just a rough comparison to learn the gap between close-set and open-set methods:

It can be observed that there is still a gap between our model and close-set baselines. Even on the Deeper dataset, where our model achieves the highest cross-dataset performance, the performance of A&B is about 6% higher than our methods and ICT even achieves 0.99. However, our effort to apply Deepfake detectors in open-set scenes repre-

sents a solid leap forward.

3. Compression and Noise Settings

3.1. Image compression settings

We adopt the JPEG compression function provided by OpenCV. The compression factor of the function is between 0 and 100, where factor 100 means no compression and factor 0 means full compression. In other words, higher factor values indicate less compression intensity. In our experiments, we set compression factors from 95 to 5 by step 5 (19 factors in total). Together with the raw image (compression factor=100), we have 20 compression degrees and the samples of different degrees can be seen in Fig. 1.

3.2. Additive noise settings

In our experiments, the additive noise is Gaussian noise. We control the intensity of additive noise, we add zero-mean Gaussian noise of different standard deviation (std) values. We set std from 1 to 19 (19 degrees in total), where higher std values indicate higher noise degrees. Together with the raw image (degree=1), there are 20 degrees of noise types. The degree of noise indicates the std of corresponding Gaussian noise. The samples of different degrees can be seen in Fig. 2.

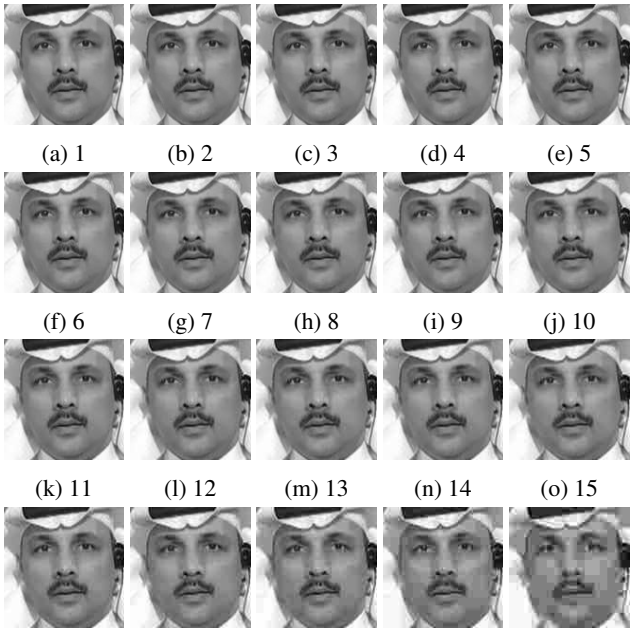

Figure 1: Samples of compressed images from compression degree 1 to degree 20.

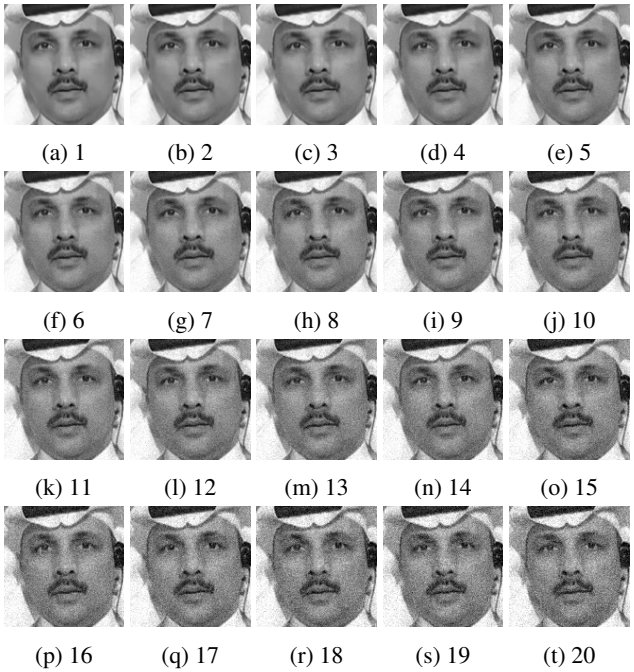

Figure 2: Samples of images with noise from noise degree 1 to degree 20.

References

[1] Shruti Agarwal, Hany Farid, Tarek El-Gaaly, and Ser-Nam Lim. Detecting deep-fake videos from appearance and behavior. In *2020 IEEE international workshop on information forensics and security (WIFS)*, pages 1–6. IEEE, 2020.

[2] Xiaoyi Dong, Jianmin Bao, Dongdong Chen, Ting Zhang, Weiming Zhang, Nenghai Yu, Dong Chen, Fang Wen, and Baining Guo. Protecting celebrities from deepfake with identity consistency transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9468–9478, 2022.