Low Resolution Fake Video Detection

Lukas Hoellein
Technical University of Munich
lukas.hoellein@tum.de

Anna Mittermair
Technical University of Munich
anna.mittermair@tum.de

Abstract

Existing approaches for face forgery detection test a video frame-by-frame without using (temporal) connections between the frames. This makes it particularly hard to detect forgery when the videos are of low resolution (e.g. highly compressed). We propose a new way to sample the videos and incorporate them into networks that make use of the temporal context between frames. We also propose architectures that make use of the optical flow between frames. Our approaches improved the detection accuracy across all four tested types of fake methods.

1. Introduction

The emergence of constantly improving methods for creating fake videos such as face-swapping or face-reenactment makes the detection of fake videos increasingly important.

State-of-the-art techniques like [3, 8, 9, 7, 13, 12] show good results on many types of fake videos but consistently have problems on low-resolution videos [8]. One reason for this is the fact that they only use single frames for determining a videos authenticity. This leads to bad detection performance for low resolution videos as the resolution reduces the amount of information contained in one frame.

One way to address this problem is to use multiple frames at once to allow detection of artifacts and temporal inconsistencies between frames. We developed and analyzed several architectures using multiple frames of a video and their temporal context to improve fake video detection for low resolution video.

2. Methods

We propose two different neural network architectures, the temporal encoder network (with or without warped images) and a network using optical flow. Both are trained on our custom version of the FaceForensics dataset [8], which we use to train our own baseline, as well.

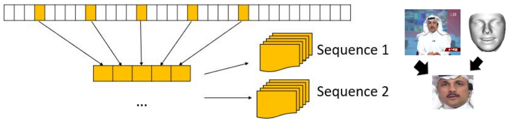

Figure 1. Extraction of sequences from a video with face crop. Each sequence has skip rate 5 and contains 5 frames.

2.1. Dataset

The FaceForensics dataset [8] offers 1000 videos, each manipulated with four different fake methods (Deepfakes (DF) [1], Face2Face (F2F) [11], FaceSwap (FS) [2] and NeuralTextures (NT) [10]) and the original videos. We only use the highly compressed, low-resolution videos.

Similar to FaceForensics [8], we sample frames from the videos. To capture the temporal context in videos, we propose to sample *sequences* of adjacent frames with a constant skip rate. Figure 1 visualizes this process. After sequence extraction, we apply a face-crop to each frame by using the NeuralTextures masks provided in [8]. Finally, we save the frames to disk in a logical file structure.

Sampled sequences vary in the number of videos, sequences per video, frames per sequence and skip rate. When using all 1000 videos, we split the sequences in training, validation, and test data according to the splits in [8]. For smaller subsets, e.g. 100 videos, we chose custom splits.

2.2. Baseline

We retrained the XceptionNet architecture from FaceForensics [8] on the sampled sequences as baseline for comparison to our other networks. It classifies every image independently, thus not using any temporal information contained in the sequences. We only retrained the last layer for binary classification, the rest is a pretrained XceptionNet on ImageNet. We used the same hyperparameters as proposed in [8] and got similar accuracy results (see Table 1).

2.3. Temporal Encoder

Our first network, the temporal encoder, makes use of the temporal context in a sequence of images. It combines multiple images together in a temporal encoder block and

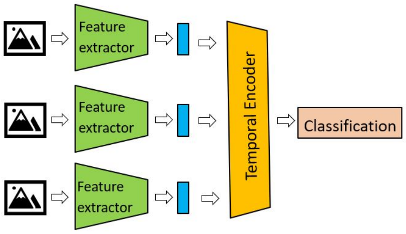

Figure 2. Architecture of the temporal encoder network: feature extraction for every image frame independently followed by concatenation of multiple image features in a temporal encoder block before binary classification.

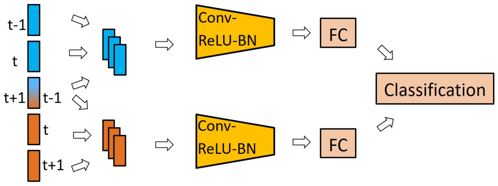

Figure 3. The temporal encoder block concatenates an amount Δ_t of image features (here Δ_t=3) and forwards them through n CNN blocks (same convolution, ReLU, batchnorm). This is done multiple times with the same layers until all input images from one sequence are processed. Finally, we combine everything with fully connected layers.

learns the binary classification task based on this input. Intuitively, this allows for detection of inconsistencies in fake videos that are no longer detectable in a single-image approach due to the low resolution. The general architecture is visualized in Figure 2.

Image feature extraction is performed by the same XceptionNet as in [8] (pretrained on ImageNet), but only using its convolution part. The temporal encoder block is shown in more detail in Figure 3. The network can be configured through the parameters Δ_t, the number of feature channels and the number of CNN blocks.

2.4. Optical Flow and Warp

Our other two architectures directly incorporate temporal information in the detection process, one by using warped frames and one by using optical flow frames. For both architectures, dense optical flow is generated using OpenCV's [4] Farneback method, which generates an optical flow tensor with size $(w \times h \times 2)$ containing the x and y components of the optical flow for each pixel.

2.5. Temporal Encoder with Warp

For using warp, we used an architecture inspired by [6]. For every sequence of frames, every frame is warped to the center frame of the sequence. This allows comparing the center frame with the warped frames to detect differences possibly caused by artifacts and temporal inconsistencies. The warped frames are passed to the temporal encoder additionally to the original frames. The remaining part is identical to the temporal encoder network as shown in Figure 2.

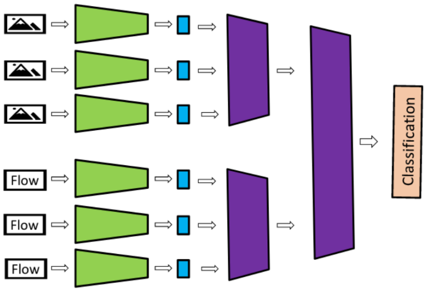

Figure 4. Fake detection architecture using optical flow. The upper branch extracts features from frames which are then concatenated along a new dimension and passed through three CNN blocks. The lower branch does the same for the optical flow tensors. Then both outputs are concatenated in the channel dimension and passed through two more convolution blocks before classification.

2.6. Architecture using Optical Flow

To directly use optical flow without warping we used an architecture similar to the Two-Stream 3D-ConvNet used for action recognition tasks in [5]. The structure of the network can be seen in Figure 4. The optical flow is calculated for pairs of images and then extended by one channel for angle of the flow vector.

3. Results

We show our quantitative results (detection accuracy) and ablation studies. All training and test results (including chosen hyperparameters) as tensorboard files, as well as all trained models can be found here: https://gitlab.lrz.de/mitterma/adl4cv

3.1. Quantitative Results

We trained our baseline and the temporal encoder network on the following dataset: 100 videos per fake method, 5 sequences per video, 10 frames per sequence, skip rate of 5, training on one specific fake method and original videos. The test accuracies for every fake method are shown in Table 1. We can see improvements of **3-15%** (percentage points) in accuracy for the same fake method as the network was trained on. Additionally, we improve up to **13%** on other fake methods (generalization) and **16-26%** on pristine (original) videos. These results illustrate the advantage of using temporal context compared to a single-frame approach.

Furthermore, we trained our warping network on Deepfakes (DF) and NeuralTextures (NT) with the same dataset

Method	DF	NT	FS	F2F	Pristine
Baseline DF	77.1	56.4	**52.9**	55.9	73.1
TE DF	**79.5**	**61.6**	51.8	**56.3**	**89.1**
Baseline NT	61.6	66.4	48.1	62.2	67.4
TE NT	**66.9**	**81.3**	**53.6**	**75.0**	**92.1**
Baseline FS	54.88	48.92	63.57	45.7	62.9
TE FS	**68.8**	**50.9**	**71.4**	**47.3**	**82.8**
Baseline F2F	60.8	60.9	**51.1**	66.3	66.3
TE F2F	**66.9**	**73.2**	50.89	**69.6**	**92.2**

Table 1. Columns: test accuracies for each (original, fake-method) pair and for original (pristine) only. Rows: baseline and temporal encoder network (TE) trained on 100 videos of one specific fake method (DF, NT, FS, F2F) and original videos. The columns of fake methods that the networks are not trained on shows their ability for generalization.

Method	DF	NT	FS	F2F	Pristine
TE DF	79.5	**61.6**	51.8	56.3	89.1
Warp DF	**81.3**	60.7	**64.3**	60.7	**93.8**
OF DF	80.0	60.0	53.0	**62.0**	80.0
TE NT	**66.9**	**81.3**	**53.6**	**75.0**	**92.1**
Warp NT	54.5	$\sim$50.0	$\sim$50.0	$\sim$50.0	$\sim$50.0
OF NT	60.0	74.0	50.0	62.0	82.7

Table 2. Test accuracies for the warp and optical flow (OF) networks trained on Deepfakes (DF). Training problem for the warp network with NeuralTextures (NT).

configuration. The results are shown in Table 2. While we can improve in the first case, the network failed to learn in the second case. This could indicate that we need more than 100 videos as dataset or that it is hard to train such a network in general. However, the good results in the first case could indicate that the additional warp input helps in generalizing to other fake methods.

The architecture using optical flow was trained on the same dataset but uses only five frames per sequence. Similarly to the other two approaches, it also performed better than the baseline on all test sets containing fake videos. However, it had worse results on the pristine test set than all other methods, which shows a problematic tendency to misclassify real videos as fake. Looking at the results in Table 2, apart from the Face2Face dataset the Temporal Encoder with warp was always superior.

3.2. Ablation Studies

We compared different configurations of the dataset generation process and the temporal encoder network. We tested values for the skip rate in sequences (see 2.1), as well as for the Δ_t parameter (see 2.3). The results are shown in Table 3. Thus, we conclude that skipping frames is actually better for capturing the temporal context than using directly adjacent frames. Also, it is better to use more frames simultaneously (larger value of Δ_t).

Skip rate Δ_t	1 3	5 3	5 5	5 7
Val. Accuracy	75.8	66.6	76.6	**86.4**

Table 3. Validation accuracies for different configurations (see 2.1, 2.3). Note that a higher Δ_t parameter for a skip rate of 1 is not possible, because in that case we had to use less frames per sequence.

Opt. flow pairs	$f_i \rightarrow f_{i+1}$	$f_i \rightarrow f_{\text{center}}$	$f_i \rightarrow f_0$
Val. Accuracy	73.9	73.4	71.7

Table 4. Possible pairings of frames for optical flow with validation accuracies (trained and tested on the DF (skip 5, sequence length of 5). Taking the optical flow from each frame to the next in the sequence is slightly better than the alternatives.

Additionally, we tested several configurations for the optical flow, with the results in Table 4 showing that calculating the optical flow between neighboring frames was the best option, leading to the results shown above.

Finally, we tested the performance of the temporal encoder network when trained on all fake methods simultaneously and when trained on all 1000 videos. However, we did not achieve good test results for the following reasons. First, training on all fake methods with only 100 videos each is too little data for our networks. We overfitted on 80% accuracy, thus the network always predicts 'fake' and is right in 4 of 5 cases since we have 4 times as much fake data than original data. We implemented a weighting between fake and original videos in the loss function, but still could not train the network. Training on more than 100 videos could possibly fix this problem. Second, we have no access to a GPU that allows training on all 1000 videos with a high enough batch-size. Training on Google Colab is not possible with the complete dataset of 1000 videos, because it takes about 9 hours just to *load* the data (not to train). Training on a local GPU was only possible with a batch-size of 4 and gave no good results either (also overfitting).

4. Conclusion

We propose a novel way to incorporate the temporal context of videos into face forgery detection. For that, we sample sequences of images from the FaceForensics dataset [8]. Our temporal encoder network improved the detection accuracy of low-resolution videos across all four fake methods and showed the advantages of using temporal context compared to single-frame approaches. Our other methods using warped images or optical flow showed promising results, but failed to train in some cases. Future work includes training the networks on the complete dataset with a powerful GPU and comparing these results to [8].

References

[1] Deepfakes github. https://github.com/deepfakes/faceswap, Accessed: 01.02.2020.

[2] Faceswap github. https://github.com/MarekKowalski/FaceSwap/, Accessed: 01.02.2020.

[3] Darius Afchar, Vincent Nozick, Junichi Yamagishi, and Isao Echizen. Mesonet: a compact facial video forgery detection network. In *2018 IEEE International Workshop on Information Forensics and Security (WIFS)*, pages 1–7. IEEE, 2018.

[4] G. Bradski. The OpenCV Library. *Dr. Dobb's Journal of Software Tools*, 2000.

[5] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6299–6308, 2017.

[6] Mengyu Chu, You Xie, Laura Leal-Taixé, and Nils Thuerey. Temporally coherent gans for video super-resolution (tecogan). *arXiv preprint arXiv:1811.09393*, 2018.

[7] Yuezun Li, Ming-Ching Chang, and Siwei Lyu. In ictu oculi: Exposing ai created fake videos by detecting eye blinking. In *2018 IEEE International Workshop on Information Forensics and Security (WIFS)*, pages 1–7. IEEE, 2018.

[8] Andreas Rossler, Davide Cozzolino, Luisa Verdoliva, Christian Riess, Justus Thies, and Matthias Nießner. Faceforensics++: Learning to detect manipulated facial images. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1–11, 2019.

[9] Joel Stehouwer, Hao Dang, Feng Liu, Xiaoming Liu, and Anil Jain. On the detection of digital face manipulation. *arXiv preprint arXiv:1910.01717*, 2019.

[10] Justus Thies, Michael Zollhöfer, and Matthias Nießner. Deferred neural rendering: Image synthesis using neural textures. *ACM Transactions on Graphics (TOG)*, 38(4):1–12, 2019.

[11] Justus Thies, Michael Zollhofer, Marc Stamminger, Christian Theobalt, and Matthias Nießner. Face2face: Real-time face capture and reenactment of rgb videos. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2387–2395, 2016.

[12] Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Guilin Liu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. Video-to-video synthesis. *arXiv preprint arXiv:1808.06601*, 2018.

[13] Xin Yang, Yuezun Li, and Siwei Lyu. Exposing deep fakes using inconsistent head poses. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 8261–8265. IEEE, 2019.