https://doi.org/10.2352/ISSN.2470-1173.2020.4.MWSF-116

Paper: Detecting "DeepFakes" in H.264 Video Data Using Compression Ghost Artifacts

Raphael Antonius Frick, Sascha Zmudzinski, Martin Steinebach
Fraunhofer Institute for Secure Information Technology SIT Darmstadt, Germany

Abstract

In recent years, the number of forged videos circulating on the Internet has immensely increased. Software and services to create such forgeries have become more and more accessible to the public. In this regard, the risk of malicious use of forged videos has risen. This work proposes an approach based on the Ghost effect knwon from image forensics for detecting forgeries in videos that can replace faces in video sequences or change the mimic of a face. The experimental results show that the proposed approach is able to identify forgery in high-quality encoded video content.

Introduction

Videos are often used on news sites or in TV news in order to support or prove the "story" in a news article or report. Nowadays, also *user created content* is increasingly provided to news channels, newspapers and news agencies. For professional journalists that raises the challenge of assessing the credibility of the content in terms of its authenticity and integrity.

Special interest in this work is on video content that features persons with their mimic and gestures presented. Tools like the (freely available) *FakeApp* or *MyFakeApp* can create specifically manipulated videos in which faces or their respective facial expressions are forged ("facial forgeries"). Such fake videos can be used for identity theft, cyber mobbing, creating fake news and even creating diffarmatory porn videos. To address this challenge, we introduce a new approach based on the analysis of compression artifacts in the spatial domain to detect facial forgeries in H.264 encoded videos.

Structure of this work

The remainder of this paper is organized as follows: In the next section, the threats on image integrity by *DeepFake*, *FaceSwap* and *Face2Face* algorithms are presented. Then an introduction to the so-called "ghost" artifacts as technical background of this work is given. The state-of-the-art in forgery detection techniques is then reviewed. The proposed algorithm is then explained in detail, followed by our experimental results. The paper concludes with a discussion of said results and displays proposals for future work.

Methods for Creating Facial Manipulations in Video Sequences

The algorithms for creating such a forgery can be divided into two categories depending on how the face of a target person in a video is altered:

Facial replacement

These techniques swap a target person's face featured in the source video with a different person's face. The facial expression / mimic of the replaced face will remain unmodified after the forgery process.

Very popular is the so-called *DeepFake* algorithm: it transfers facial textures by utilizing deep neural network methods. Currently three actively maintained open-source implementations of the algorithm exist [1, 2, 3]. Since a model needs to be learned before execution, it is not possible to execute the algorithm in real-time.

Another approach was presented by *Kowalski* et al. under the project name "FaceSwap" [4]. It can process video data in real-time.

Facial reenactment

By contrast, reenactment adapts the mimic of the target person to the facial expression of the attacker. The latter acts as a "source actor" who features the intended face expression. The target face itself is not altered and therefore still features the same person in the video. One example is the *Face2Face* algorithm by *Thies* et al. [5]

Background – The Ghost Artifact

The proposed video forensic approach is based on the so-called *ghost artifacts*. It was introduced by *Farid* [6] for (still) image data in JPEG format. It is one among several approaches to exploiting the compression error for forensic purpose. The analysis is conducted in the *spatial* domain, i.e. on the decoded RGB pixel values.

Here, the following phenomenon can be observed: In the standard JPEG compression scheme, image data is represented in terms of quantized spectral DCT-coefficients. Larger quantization step sizes result in a better compression ratio and thus in an overall lower image quality (and vice versa).

Let $\mathscr{C}$ be the set of DCT-coefficients in an arbitrary image I which were quantized using a quantization table. The values of the quantization table are based on a given quality parameter, i.e. the quantization value q. Let further assume that I is consecutively compressed a *second time* using quantization value q'. The respective result is denoted as $\mathscr{C}'$. It was shown by the original authors that the difference between the sets of DCT-coefficients $\mathscr{C}$ and $\mathscr{C}'$ will be minimal when $q = q'$.

This can be exploited for detecting splicing in images and videos, respectively. Let us assume that a forged image I_1 is created by inserting image data from a different source image I_0 (respective quantization step size q_0) into parts of the original authentic image content. The image I_1 is saved using quantization

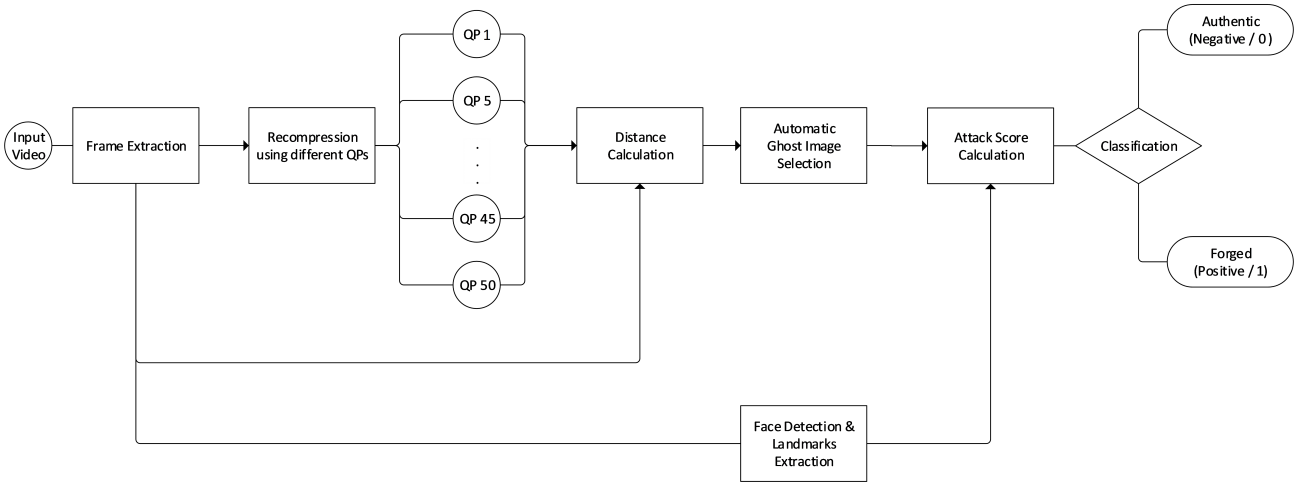


Figure 1: Proposed Analysis Pipeline using Ghost Artifacts

step size q_1 (assumed: is $q_1 < q_0$) to obtain $\mathscr{C}_1$. Hence, the forged image data features DCT-coefficients that were subject to quantization with using step sizes both q_1 and q_0.

Now, for test purpose let us recompress I_1 a second time using different settings of quantization size q_2, resulting in $\mathscr{C}_2$. As stated above, the difference between $\mathscr{C}_1$ and $\mathscr{C}_2$ will be minimal, when q_2 is chosen such that $q_2 = q_1$. However, since $\mathscr{C}_2$ contains data initially quantized using a quantization value of q_0, a second minimum will be exposed when q_2 is chosen such that $q_2 = q_0$! The second minimum is referred to as the *JPEG ghost*.

Based on this, *Farid et al.* presented a non automatic image splicing detection method. For investigating an input image, it is temporarily recompressed using varying quantization parameters. Then, the respective averaged difference is calculated using all three colors in the RGB color space. For those quantization parameters that match either the *authentic* image content or the *forged* facial area, the respective local difference becomes nearly zero. Hence the forged area can be revealed.

Although the different life cycle with respect to (temporary) re-encoding affects the respective DCT in the spectral domain the detection method effectively analyses in the spatial / pixel domain.

However, it is only able to detect traces of a forgery in cases, where $q_1 < q_0$, i.e. the image quality is not further decreased when saving the forged image.

Related Works

In this paper, we base our approach on the ghost artifacts proposed by Farid et al. [6] as explained above. This method locates tampered regions by analyzing artifacts introduced into an image due to double compression. Further approaches based on double compression artifacts were presented in [7, 8, 9]. Other approaches are based on inconsistencies in the traces caused by sensor pattern noise, the color filter array or the chromatic aberration [10, 11, 12].

Furthermore, some methods to detect especially facial forgeries in video sequences exist. Works presented in [13, 14, 15] utilize domain specific features for the tampering detection, e.g. inconsistencies in eye blinking, and are therefore limited to the detection of DeepFake videos.

In further works approaches for a more generalized classification are proposed: *Fridrich* and *Kodovsky* [16] use steganalysis features and support vector machines (SVM). *Roessler, Cozzolino et. al.* [17] deploy a neural network instead of SVM for classification. Also works featured by *Rahmouni et al.* [18], *MesoNet* by *Afchar et al.* [19] and *XceptionNet* by *Chollet et al.* [20] deploy specialized convolutional neural networks for classification.

An extended overview about related works specializing in the detection of DeepFakes can be found in [21].

Proposed Method

Applying Ghost Effect on Video Frames

Our method takes advantage of the ghost artifacts (see Section). Here, it is applied on video in terms of its (still) image data in the video frames. According to the original ghost algorithm [6] we propose to recompress the input video using varying quantization parameters q. Then, the "difference" between the input video frames and the newly compressed (temp) video frames are calculated in the spatial domain. Areas, which were compressed prior to the final compression using the same q value as the recompressed video feature the ghost artifact. However, this assumption is only true if the compression rate of the final compression is lower than the compression rates of previous compressions.

Detection Algorithm

Let us assume that an adversary subsequently uses e.g. the *DeepFake* system for applying a facial forgery. He/she eventually re-encodes the forged single frames to a doctored video (at quality q_1) and distributes it. The proposed detection scheme can be outlined as follows (Figure 1):

1. *Input:* The doctored video is input to our forensic method.
2. *Video Decoding & Frame Extraction:* At first, the input is decoded and its frames of size $M \times N$ are extracted. For simplicity, only I-frames are utilized. The I-frame data is maintained for later comparison in step 4) below.
3. *Re-compression:* In order to trigger the ghost artifacts, the extracted frames are merged into a video and then they are recompressed using a varying quantization parameter $q_2 \in S$, i.e. at different quality settings $S = \{1, 5, 10, 15...50\}$, as long as $q_2 \neq q_1$ applies.

The latter condition reflects that in the case of $q_2 = q_1$, the difference between those two images would be equal (almost) completely and forged areas could not be identified. However, that condition can be relaxed on real-world H.264 data as the quantization parameter for each macro block does not need to be constant throughout the whole frame, anyway.

4. *Distance Calculation:* Then the "difference" for each of the three color channels between the input video frame $I(x,y)$ and the recompressed temp video frames at quality parameter q (denoted as $I_q(x,y)$) is defined as follows:

$$G_q(x,y) = |I(x,y) - I_q(x,y)| \ , \ q \in S \ \wedge \ q \neq q_1 \qquad (1)$$

Then, each of the ten intermediate results is binarized using a threshold value of $t = 20$ for each color channel $c \in \{R,G,B\}$. This expedites eliminating outliers and reducing the false positive rate. We obtain the binarized difference image

$$B_{p,q}(x,y) = \begin{cases} 1 & \text{if } \exists\, c : G_q(x,y)[c] > t \quad \forall\, c,(x,y) \\ 0 & \text{else} \end{cases}$$

in which the pixels of B serve as "flags" indicating the ghost effect (or not, resp.).

5. *Face Detection & Facial Landmarks Extraction:* In parallel, faces that are present in the video frame are detected by means of *facial landmarks* extraction. The region inside the bounding box will serve as an "image patch" of size $M' \times N'$ used for classification. The facial landmarks help identifying chin and forehead region which allows refining the classification.
6. *Ghost Image Selection:* The video frame which actualla features *ghost* artifacts is selected as follows: we define the ratio of allocated 1-flags as

$$R_q = \frac{\sum_{x,y} B_q(x,y)}{M \cdot N} \quad q \in S \ \wedge \ q \neq q_2 \qquad (2)$$

In most cases, the background covers the greater part of an image. Hence, the image with the lowest ratio $\min_q \left(R_q\right)$ value is identified as the actual ghost image and shall be selected for the classification.

7. *Attack Score Calculation:* Similar to Equation 2, in the identified ghost image the attack score is computed *for the respective $M' \times N'$ facial area only* for each extracted face.

$$R'_q = \frac{\sum_{x,y} B_q(x,y)}{M' \cdot N'} \quad q \in S \ \wedge \ q \neq q_2 \qquad (3)$$

Forged faces will feature an attack score R' significantly greater than in any non-doctored image due to the missing ghost effect in the actually doctored facial areas.
Eventually, a reasonable threshold is then used to classify, whether the face is authentic or not. The steps above are carried out for all faces found inside a frame.

It should be noted that when recompressing a video as explained before, the encodes shall be configured such that macro block assignment and hence the motion-vectors shall be maintained as much as possible, see [22].

Example

Figure 2a shows a single frame from an original (i.e. authentic) video (at quality q_1). Figure 2c shows the correspondent *forged* video frame (at quality q_2). Figures 2d and 2e show binarized difference images for two temporarily recompressed video frames at different q values, as explained in step (3) above. Figure 2d features many areas that are rather bright. By contrast, the image in Figure 2e is mostly rather dark: it features *ghost* effect, i.e. the correct value q_1 was successfully estimated. In this image, forgery is successfully detected inside the bounding box (marked as red rectangle) of the news anchor's face. For the correspondent authentic image (Figure 2b), the respective facial area remains rather dark.

Potential *false positives* become observable mostly when object edged at high contrast are present. Thanks to restricting the classification to the facial bounding box in the first place, these areas will be ignored in our approach.

Implementation Detail

The detection algorithm was developed in *Python*. The *OpenCV* library was used for the image input and output processing [23], as well as the *FFmpeg* framework for creating the ghost videos [24]. The face alignment and detection library *face-alignment* [25], based on *Dlib* [26], was used to extract the facial landmarks.

Experimental Results

Evaluation Dataset

For the calculation of the classification threshold value and evaluating the proposed method, the *FaceForensics++* dataset was used [27]. The dataset contains 1000 videos for each forgery algorithm (*DeepFake*, *FaceSwap* and *Face2Face*) as well as the original videos (size: VGA up to full HD, single and multiple faces visible). The origin of the authentic videos is the *YouTube-8M dataset* [28].

Our dataset provides the videos in different compression rates. For our experiments, the "raw" videos were analyzed. These videos are not truely raw as their origin often lie in already compressed Youtube videos. However, they have been compressed with a quantization parameter of "0" after conducting the forgery (maximum H.264 quality). After the forgery, the videos have been saved using the same quantization parameter again.

For the threshold value selection, the *FaceForensics++* was divided into a training and a test set. The training set consists of the first 100 respective original, *DeepFake*, *FaceSwap* and *Face2Face* videos. The test set consists of the remaining 900 videos.

Search Window & Threshold Selection (Training Set)

From closer analysis a threshold value $R = 0.028$ was identified as feasible (in terms of achieving the equal error rate) for classification during the experiments. Based on this selection the following results on performance criteria could be observed:

- Accuracy = 0.9716, Precision = 0.9799, Recall = 0.9801,
- F1-Score = 0.9800.

The reader is reminded that the F1-score is a suitable performance measurement in cases, where the class distribution is unbalanced.

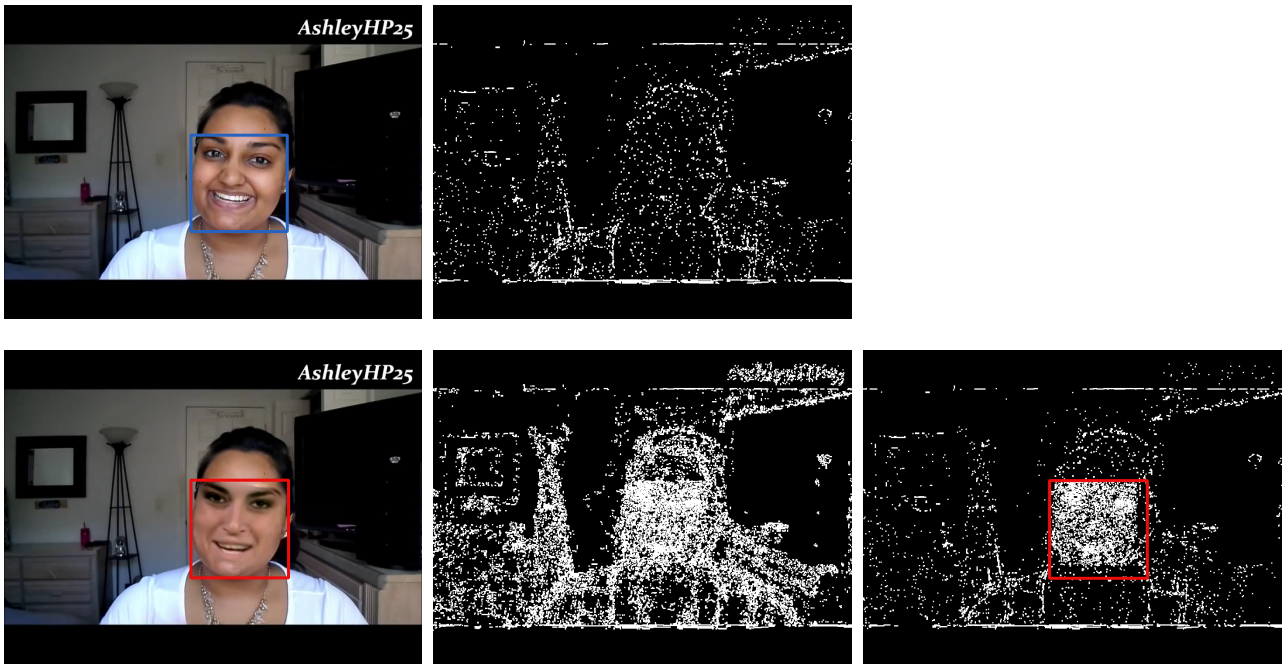


Figure 2: Authentic and Forged Video Frames and their Corresponding Binarized Difference Images
a). Authentic Frame **b).** Authentic Diff Image vs. q=1
c). *DeepFake*d frame **d).** Forged Diff Image vs. q=10 **e).** Forged Diff image vs. q=1

The latter is the case in our experiments as we evaluate three times as much *forged* than correspondent *authentic* videos (i.e. from the three different kind of attacks).

Classification Results on the RAW Compressed (Test Set)

In order to verify the performance of the proposed method, the algorithm has been applied on 900 RAW compressed videos of the *FaceForensics++* dataset. The length of each video ranges between *10* to *15* seconds. The experiments were conducted on a virtualized Intel Xeon E5-2687W @ 3.00GHz using 24 cores. Depending on the length of the video, and whether all frame-types or just I-frames are classified, the classification process can take between several seconds to some minutes. By optimizing the code to fully utilize multiple threads, these timings might greatly improve in the future.

At first each of the threes facial forgery algorithms has been evaluated individually and then the overall classifier performance is expressed in terms of the *area under curve (AUC)* value.

From the values provided in Figure 3 it can be seen that the area under ROC curve is calculated AUC to be 0.99 for all three attack types. Accuracy of the respective combination of the three single results is 97.16%, F1-score is found to be 0.9754 and correspondent *AUC* = 0.99. This means, that the model is able to classify forged from authentic facial quite well. Classification performance is almost equally good across the three forgery methods.

Further analysis shows that the performance of the proposed algorithm depends on several factors, e.g. the facial detection accuracy of the used face detection algorithm and the size of the area, which is actually modified by the forgery algorithm.

One issue was found because when CGI content or footage using *chroma-keying* ("green screen") was analyzed: if such inserted present it will not exhibit the ghost effect because of its different "compression lifecycle". Hence, in those areas the estimated ratio R' will remain high, hence actually authentic faces could be misclassified as forged.

Further analysis on the *FaceForensics++* data set at medium and low visual quality, e.g. at $q = 23$ and $q = 40$ showed that our classifier is not able to correctly classify forged videos in medium or low quality videos.

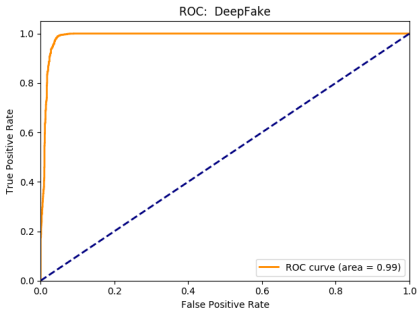
Moreover, further analysis showed our approach has difficulties in correctly distinguishing authentic from forged videos, when misalignment of the 8x8 or 4x4 DCT-grid occurs during post processing (rescaling, rotation). This behavior is typical for detection schemes based on compression artifacts.

Discussion and Comparison to State-of-the-Art

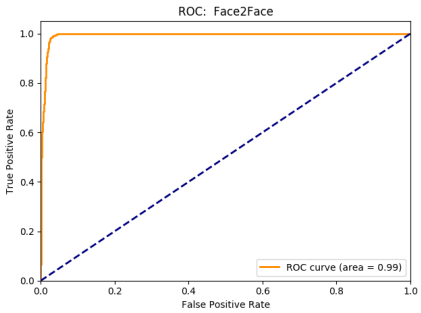
In a work by Roessler et al. [27], a survey was conducted with 143 human participants. RAW encoded *DeepFake*, *FaceSwap*, *Face2Face* and authentic videos were classified manually with an accuracy of about 75%. The participants had the most difficulties in correctly identifying *Face2Face* videos with an accuracy of only 41.93%. With the increment in compression, the correctly classification rate drops to a value below 67.69% in the case of *DeepFake*, *FaceSwap* and pristine videos and 43.13% in the case of *Face2Face* videos.

Furthermore, in the published paper of the *FaceForensics++* dataset, several existing methods for detecting face forgeries have been evaluated.

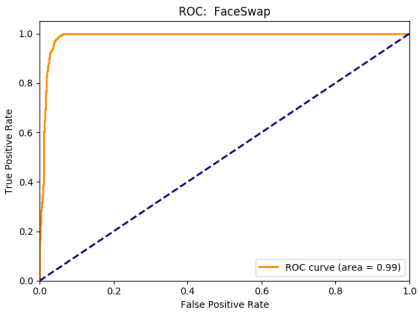
As it can be seen in Table 1, for RAW ($q = 0$) compressed videos the state-of-the-art algorithms perform with an accuracy of 98.03% to 99.28% in the case of *DeepFakes*, 97.96% to 99.61%



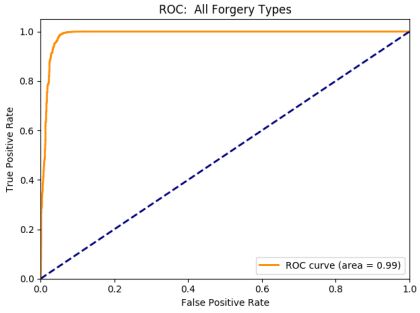
Accuracy = 0.9684 Precision = 0.9779,
Recall = 0.9571, F1-Score = 0.9674



Accuracy = 0.9759, Precision = 0.9970,
Recall = 0.9547, F1-Score = 0.9754



Accuracy = 0.9621, Precision = 0.9625,
Recall = 0.9523, F1-Score = 0.9574



Accuracy = 0.9716, Precision = 0.9799,
Recall = 0.9801, F1-Score = 0.9800

Figure 3: ROC-Curve of Various Classifications (RAW Videos)
a). DeepFake (AUC = 0.99) **b).** Face2Face (AUC = 0.99) **c).** FaceSwap (AUC = 0.99)
d). All Forgery Types (AUC = 0.99)

	Fridrich [16]	Cozzolino [17]	Rahmouni [18]	MesoNet [19]	XceptionNet [20]	our method
DeepFake	99.03%	98.83%	98.03%	98.41%	99.06%	96.84%
Face2Face	99.13%	98.56%	98.96%	97.96%	99.61%	97.59%
FaceSwap	98.27%	98.89%	98.94%	96.07%	99.14%	96.21%
All Combined	97.63%	98.56%	97.72%	96.51%	99.41%	97.16%

Table 1: Accuracies (RAW Videos) of our proposed method and related work

for *Face2Face* videos and *FaceSwap* videos with an accuracy of 98.27% to 99.14%. Our proposed method based on ghost artifacts is able to achieve an accuracy of 98.40% for *DeepFake* videos, 98.64% in the case of *Face2Face* videos and an accuracy of 98.03% in the case of the classification of *FaceSwap* forged videos.

However, the proposed method is not able to classify HQ ($q = 23$) and LQ ($q = 40$) videos well. It was only able to classify HQ videos with an accuracy of 65.13% and an AUC score of 0.61 and LQ videos with an accuracy of 54.18% and an AUC score of 0.63 respectively.

Hence, in terms of the classification on RAW compressed videos, the proposed method is as performant as currently available facial forgery detection algorithms. Like other related works, it indeed outperforms *manual* human inspection by far.

XceptionNet is the only method, which is able to successfully classify low quality videos compressed with a q of 40 with an accuracy of over 90% of face replacement algorithms and 89.8% for *Face2Face* videos. Since the method of *Fridrich et al.* and *Cozzolino et al.* uses features gathered from high-pass filtered images, the algorithm suffers from the same issues for lower video quality content as the proposed method of our work.

Conclusions and Future Work

In this paper, an alternative approach to detect *facial* forgeries in H.264 encoded videos has been proposed. This method is based on the "JPEG ghost" algorithm, which detects tampered faces by analyzing for inconsistent compression errors in the authentic versus the tampered image regions.

Tests of the proposed method were carried out using the *FaceForensics++* dataset. The results show, that the proposed approach is able to classify high-quality encoded *DeepFake*, *FaceSwap* and *Face2Face* test videos. We achieve a high classification performance in terms of accuracy values, F1 scores and correspondent area under ROC curve (AUC values). The proposed method can be applied to locate different kinds of facial forgeries in H.264 videos, which exhibit compression artifacts.

Use cases for our works are scenarios in which it can be assumed that the high quality version of a video in question can be made available for verification. Examples are in the field of pro-

fessional "producers" of news, i.e. news agencies, broadcasters, journalists / reporters.

Future work could extend this method to detect general video splicing attempts (beyond *DeepFakes* and the like), to improve the accuracy for video content featuring lower visual quality settings and to measure how the detection rate is affected by the use of different H.264 codecs.

Acknowledgement

We would like to thank the "Visual Computing Group" at TUM University, Munich, Germany, for providing access to the *FaceForensics++* dataset [27].

This work was supported by the German Federal Ministry of Education and Research (BMBF) under grant agreement "Lernlabor Cybersicherheit".

References

[1] deepfakes (Github user). (2019) Github repository 'deepfake/faceswap'. [Online]. Available: https://github.com/deepfakes

[2] iperov (Gitlab user). (2019) DeepFaceLab, Gitlab repository 'iperov/DeepFaceLab'. [Online]. Available: https://github.com/iperov/DeepFaceLab

[3] shaoanlu (Github user). (2019) Github repository 'shaoanlu/faceswap-GAN'. [Online]. Available: https://github.com/shaoanlu/faceswap-GAN

[4] M. Kowalski. (2016) FaceSwap (Gitlab repository). [Online]. Available: https://github.com/MarekKowalski/FaceSwap

[5] J. Thies, M. Zollhöfer, M. Stamminger, C. Theobalt, and M. Nießner, "Face2face: Real-time face capture and reenactment of rgb videos," *Commun. ACM*, vol. 62, no. 1, pp. 96–104, Dec. 2018.

[6] H. Farid, "Exposing digital forgeries from jpeg ghosts," *IEEE Transactions on Information Forensics and Security*, vol. 4, no. 1, pp. 154–160, March 2009.

[7] S. Milani, M. Tagliasacchi, and S. Tubaro, "Discriminating multiple jpeg compression using first digit features," in *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, March 2012, pp. 2253–2256.

[8] T. Bianchi and A. Piva, "Image forgery localization via block-grained analysis of JPEG artifacts," *IEEE Transactions on Information Forensics and Security*, vol. 7, no. 3, pp. 1003–1017, jun 2012.

[9] W. Li, Y. Yuan, and N. Yu, "Passive detection of doctored JPEG image via block artifact grid extraction," *Signal Processing*, vol. 89, no. 9, pp. 1821–1829, sep 2009.

[10] J. Lukas, J. Fridrich, and M. Goljan, "Digital camera identification from sensor pattern noise," *IEEE Transactions on Information Forensics and Security*, vol. 1, no. 2, pp. 205–214, June 2006.

[11] M. K. Johnson and H. Farid, "Exposing digital forgeries through chromatic aberration," in *Proceedings of the 8th Workshop on Multimedia and Security*, ser. MM & Sec '06. New York, NY, USA: ACM, 2006, pp. 48–55.

[12] P. Ferrara, T. Bianchi, A. D. Rosa, and A. Piva, "Image forgery localization via fine-grained analysis of CFA artifacts," *IEEE Transactions on Information Forensics and Security*, vol. 7, no. 5, pp. 1566–1577, oct 2012.

[13] Shallow, "Machine learning for detecting fake videos. github repository 'mvaleriani/shallow'." http://shallow-ai.com/, 2019.

[14] Y. Li and S. Lyu, "Exposing deepfake videos by detecting face warping artifacts," *arXiv*, vol. arXiv:1811.00656v3 [cs.CV], 2019.

[15] Y. Li, M.-C. Chang, and S. Lyu, "In Ictu Oculi: Exposing AI Created Fake Videos by Detecting Eye Blinking," University at Albany, State University of New York, NY, USA, Tech. Rep., 2018.

[16] J. Fridrich and J. Kodovsky, "Rich models for steganalysis of digital images," *IEEE Transactions on Information Forensics and Security*, vol. 7, no. 3, pp. 868–882, jun 2012.

[17] A. Rössler, D. Cozzolino, L. Verdoliva, C. Riess, J. Thies, and M. Niessner, "Faceforensics: A large-scale video dataset for forgery detection in human faces," *arXiv*, vol. arXiv:1803.09179v1 [cs.CV], 2018.

[18] N. Rahmouni, V. Nozick, J. Yamagishi, and I. Echizen, "Distinguishing computer graphics from natural images using convolution neural networks," in *2017 IEEE Workshop on Information Forensics and Security (WIFS)*, Dec 2017, pp. 1–6.

[19] D. Afchar, V. Nozick, J. Yamagishi, and I. Echizen, "Mesonet: a compact facial video forgery detection network," *arXiv*, vol. arXiv:1809.00888v1 [cs.CV], 2018.

[20] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," *arXiv*, vol. arXiv:1610.02357v3 [cs.CV], 2016.

[21] R. Tolosana, R. Vera-Rodríguez, J. Fiérrez, A. Morales, and J. Ortega-Garcia, "Deepfakes and beyond: A survey of face manipulation and fake detection," *ArXiv*, vol. abs/2001.00179, 2020.

[22] S. Milani, P. Bestagini, M. Tagliasacchi, and S. Tubaro, "Multiple compression detection for video sequences," in *2012 IEEE 14th International Workshop on Multimedia Signal Processing, MMSP 2012 - Proceedings*, 09 2012, pp. 112–117.

[23] OpenCV team. Open Source Computer Vision Library (OpenCV). [Online]. Available: https://opencv.org/

[24] FFmpeg. (2019) FFmpeg multimedia framework. [Online]. Available: https://ffmpeg.org/

[25] A. Bulat and G. Tzimiropoulos, "How far are we from solving the 2D & 3D face alignment problem? (and a dataset of 230,000 3d facial landmarks)," in *International Conference on Computer Vision*, 2017.

[26] D. E. King, "Dlib-ml: A machine learning toolkit," *Journal of Machine Learning Research*, vol. 10, pp. 1755–1758, 2009. [Online]. Available: http://dlib.net/

[27] A. Rössler, D. Cozzolino, L. Verdoliva, C. Riess, J. Thies, and M. Niessner, "Faceforensics++: Learning to detect manipulated facial images," *arXiv*, vol. arXiv:1901.08971v2 [cs.CV], 2019.

[28] S. Abu-El-Haija, N. Kothari, J. Lee, P. Natsev, G. Toderici, B. Varadarajan, and S. Vijayanarasimhan, "YouTube-8M: A Large-Scale Video Classification Benchmark," *arXiv*, vol. arXiv:1609.08675v1 [cs.CV], 2016.

Author Biography

Raphael Antonius Frick *is a student of computer science at the Technische Universität Darmstadt, Germany. Currently, he works as an assistant researcher at the Media Security and IT Forensics division at Fraunhofer SIT. His research is focused on detecting manipulations in image and video data.*

Dr. Sascha Zmudzinski is researcher at Fraunhofer SIT. He received PhD from Technische Universität Darmstadt for his work on watermarking based audio authentication. His activities cover different subjects in multimedia security like forensics, watermarking and perceptual hashing.

Prof. Dr. Martin Steinebach *is the Manager of the Media Security and IT Forensics division at Fraunhofer SIT. In 2003 he received his PhD at the Technical University of Darmstadt for this work on digital audio watermarking. In 2016 he became honorary professor at the TU Darmstadt.*

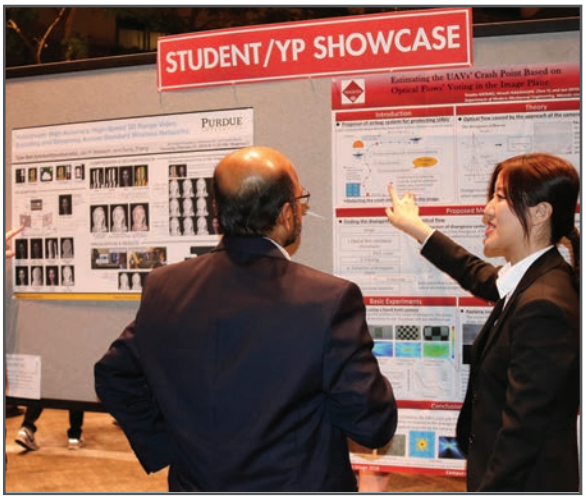
JOIN US AT THE NEXT EI!

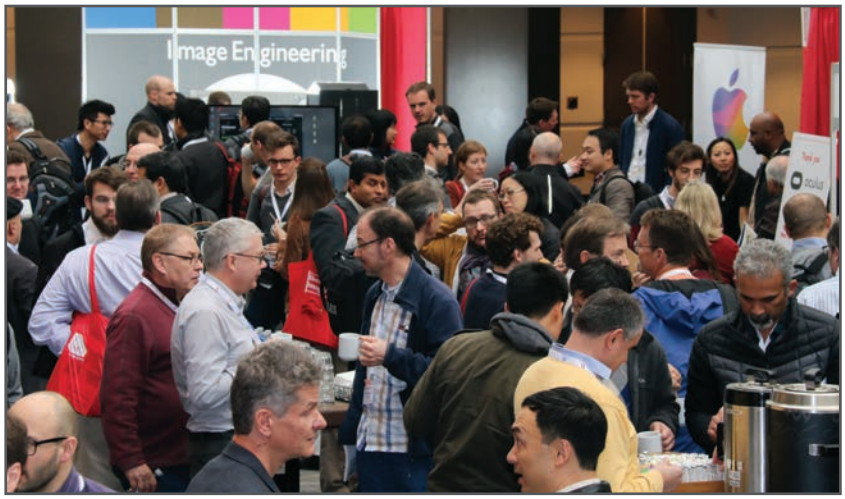
IS&T International Symposium on

Electronic Imaging

SCIENCE AND TECHNOLOGY

Imaging across applications . . . Where industry and academia meet!





• SHORT COURSES • EXHIBITS • DEMONSTRATION SESSION • PLENARY TALKS •
• INTERACTIVE PAPER SESSION • SPECIAL EVENTS • TECHNICAL SESSIONS •

www.electronicimaging.org

IS&T
imaging.org