Fooling State-of-the-Art Deepfake Detection with High-Quality Deepfakes

Arian Beckmann
Fraunhofer Heinrich-Hertz-Institute
Berlin, Germany
arian.beckmann@hhi.fraunhofer.de

Anna Hilsmann
Fraunhofer Heinrich-Hertz-Institute
Berlin, Germany
anna.hilsmann@hhi.fraunhofer.de

Peter Eisert
Fraunhofer Heinrich-Hertz-Institute
Berlin, Germany
Humboldt University of Berlin
Berlin, Germany
peter.eisert@hhi.fraunhofer.de

ABSTRACT

Due to the rising threat of deepfakes to security and privacy, it is most important to develop robust and reliable detectors. In this paper, we examine the need for high-quality samples in the training datasets of such detectors. Accordingly, we show that deepfake detectors proven to generalize well on multiple research datasets still struggle in real-world scenarios with well-crafted fakes. First, we propose a novel autoencoder for face swapping alongside an advanced face blending technique, which we utilize to generate 90 high-quality deepfakes. Second, we feed those fakes to a state-of-the-art detector, causing its performance to decrease drastically. Moreover, we fine-tune the detector on our fakes and demonstrate that they contain useful clues for the detection of manipulations. Overall, our results provide insights into the generalization of deepfake detectors and suggest that their training datasets should be complemented by high-quality fakes since training on mere research data is insufficient.

CCS CONCEPTS

• **Computing methodologies** → **Appearance and texture representations**; • **Security and privacy** → **Privacy protections**; • **Social and professional topics** → *Identity theft*.

KEYWORDS

deepfake detection, face swapping, forgery, dataset

ACM Reference Format:
Arian Beckmann, Anna Hilsmann, and Peter Eisert. 2023. Fooling State-of-the-Art Deepfake Detection with High-Quality Deepfakes. In *Proceedings of the 2023 ACM Workshop on Information Hiding and Multimedia Security (IH&MMSec '23), June 28–30, 2023, Chicago, IL, USA.* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3577163.3595106

1 INTRODUCTION

Various deep-learning frameworks enable the manipulation of visual media. Deepfakes denote images and videos that show a face whose identity or expression has been manipulated by a deep neural network. Besides fun gimmicks, deepfakes pose a threat to security and privacy. A well-known misuse of this technology is the creation of fake pornographic content showing people who did not consent to have their data used for this purpose [12]. Another malicious use of deepfakes lies in the impersonation of other identities. One could use a deepfake to bypass security precautions that rely on visual data. Moreover, deepfakes can show influential personalities, such as politicians spreading dangerous misinformation and hence pose a threat to society and democracy [29]. To generate a deepfake, deep neural networks, usually convolutional neural networks (CNNs), are utilized to extract visual information and generate novel images. We speak of the person whose face is showing in a deepfake as the "target identity" or "target", while the expression is transferred from the "driving identity" or just "driver". Information extracting CNNs, also called encoders, are used to extract the appearance information from data showing the target person. Depending on the architecture of the deepfake model at hand, the same or another encoder extracts the information on expression, and possibly pose and illumination, from the driver image. The extracted information is then fed to a decoder, usually another CNN, to generate a face image showing the target identity with the expression specified by the driver. Since the appearance of deepfakes, independent developers and researchers have been participating in an adversarial game in which one side tries to improve the visual quality and realism of the fakes while the other aims to detect those robustly. On the one hand, complex model architectures and training procedures, as well as sophisticated extensions to existing models, have been proposed to boost the visual quality of deepfakes [8, 19, 21, 22, 24, 30, 32, 33, 37]. On the other hand, deepfake detectors also employ sophisticated architectures, are extended to consider multi-modal data, and pay attention to (common) artifacts, also those not visible to the human eye. This leads to better detection performance, including increased robustness and generalizability [1, 10, 11, 17, 20, 25, 34, 38]. A bottleneck for the development of efficient detection methods lies in the limited availability of high-quality datasets for training and testing. A handful of deepfake databases are available and regularly used for the training and evaluation of detectors [7, 15, 18, 25, 39]. However, these datasets are subject to various limitations, such as a lack of variability in generation methods. A major weakness of the datasets is the lack of realism of the fakes. Poor visual quality and occasionally used blending procedures lead to visual artifacts that can be detected by the human eye. To generate a large number of fake videos showing a variety of identities, the entire generation process, including the gathering of training data, is usually fully automated. Nevertheless, this can lead to deficient training data, given that a dataset for a single person can include blurry images and even images showing the face of another person. This and

IH&MMSec '23, June 28–30, 2023, Chicago, IL, USA

ACM ISBN 979-8-4007-0054-5/23/06.
https://doi.org/10.1145/3577163.3595106

the already small amount of training data available lead to poorly trained models, which in turn generate fakes of low visual quality.

In this work, we address the problem of lacking realism in deepfake datasets. We demonstrate that models trained on common benchmarks do not generalize to well-crafted fakes. We create a new set of high-quality deepfakes, which we feed to a state-of-the-art deepfake detector that has been shown to generalize well to unseen datasets and forgery methods. The evaluation shows that the detector struggles with the detection of our fakes, while it performs outstandingly on the detection of their pristine counterparts. We then finetune the detector on our dataset and show that our fakes contain clues for fake detection, thereby highlighting the need for more well-crafted fakes in common deepfake databases. Our contribution is threefold:

- We propose a novel architecture for faceswap deepfakes that is simple to train and produces high-quality results.
- Additionally, we propose a simple extension to the common blending procedure in faceswaps which leads to fewer blending artifacts.
- We provide useful insights on the generalization of deepfake detectors as well as the necessity for well-crafted fakes in their training.

2 RELATED WORKS

2.1 Deepfake Generation

A common deepfake approach uses a dual-decoder autoencoder architecture and is trained for two specific identities [6]. With time, a variety of extensions and modifications to this framework were proposed. Overall, newer approaches extract the target appearance from merely a single or a few images in order to obtain generalization with respect to identities. Several approaches utilize adversarial training to increase the visual quality of the fakes [21, 22, 24, 30, 33]. Other works incorporate 3D morphable models [3] to encode the expression and pose information of the driver [21, 30]. Furthermore, methods that manipulate the latent space of a trained StyleGAN [13, 32, 33] were proposed. More recent works aim to generate fakes in higher resolution with strong details, but still lack high perceptive realism [8, 19, 33, 37].

2.2 Deepfake Detection

Early detection approaches focus on known clues in deepfakes such as missing eye blinking [17] or inconsistent head poses [34]. Other works utilize simple CNNs to detect (high-level) visual artifacts caused by the forgery model [20, 25] or post-processing operations like face blending [16]. Due to the increase in the visual quality of fakes, later works focus on detecting low-level artifacts [38]. A promising line of research aims to detect inconsistencies in motion, especially in movements caused by speaking [1, 10, 11], leading to stable performances in cross-manipulation scenarios.

2.3 Deepfake Datasets

The first databases containing deepfake videos were proposed to facilitate the development of deepfake detection algorithms [15, 34]. Subsequent works present benchmarks containing more fakes with better visual quality. A widely used benchmark is the FaceForensics++ (FFPP) dataset [25], which consists of 1,000 videos collected from YouTube and corresponding fakes generated by six different synthesis methods. The Celeb-DF v2 (CDF) dataset [18] provides over 5,000 fake videos and aims to overcome the issue of low visual quality in deepfake benchmarks. Currently, the largest database is part of the Deepfake Detection Challenge (DFDC) [7]. This dataset consists of more than 100,000 fake sequences generated by eight different synthesis procedures using the recordings of 960 cooperating actors. Recently published works aim to enable the training and testing of deepfake detectors with a focus on generalization to unseen manipulation methods and real-world scenarios [23, 39].

3 CREATION OF HIGH-QUALITY DEEP FAKES

This section presents our approach to the generation of deepfakes. Since we want to generate fakes with high visual quality, we use a dual-decoder autoencoder. This architecture, originally proposed by [6], allows the training of a model for two specific identities and is thus able to learn meaningful and highly detailed representations of their appearances. Once fully trained, the encoder can encode any face image of the two identities into a latent code containing extensive information about the present expression, pose, and illumination. Furthermore, each decoder can transform this code into a face image of its corresponding identity. To further improve deepfake quality, we provide modifications and extensions to the approach in [6]. We use a novel autoencoder utilizing an EfficientNet-B4 [28] as the encoder with several residual blocks in the decoder. Moreover, we propose an advanced blending procedure that produces fewer blending artifacts when merging the forged face of the target with the driver's head. Details on which datasets are used to train our models are given in section 4.

The section is structured as follows: First, we describe our data collection process, which results in one training dataset, also called faceset, per identity. Thereafter, we present our autoencoder architecture alongside its training details. The section is concluded with a description of the conversion process, which includes the proposal of our advanced blending procedure.

3.1 Faceset Collection

We now describe the faceset collection procedure, which is adapted to the one in [6]. A faceset is an identity-specific dataset of face images displaying a large variety of expressions, poses, and lighting scenarios. In order to train an autoencoder to perform a faceswap between two set identities, we require both their facesets as training data. To collect a faceset, we need to gather multiple videos or images showing the person, possibly from various resources. It is of utmost importance that the data is of high visual quality, as the model will only be able to learn fine visual details if these are visible in the training data. Once the data is gathered, we extract all faces using FAN [4]. The face detection process also returns 68 facial landmarks, which are utilized to align the head to a neutral position in the center of the image and crop the image to 512^2 pixels. Moreover, we use BiSeNet [5, 35] to obtain a segmentation of the facial regions in each image. This segmentation is essential for the training of the autoencoder, as explained in 3.2. Given the fully extracted faceset, we clean it by removing faces that are blurry, too

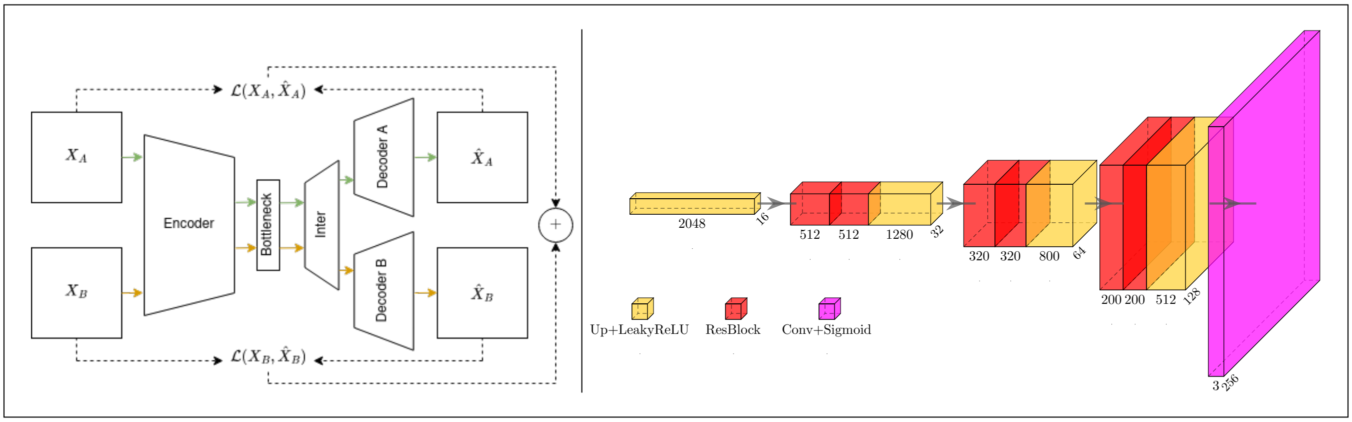

Figure 1: Left: Forward pass of our model. The loss is computed separately for each identity (A and B) and finally summed up. Right: Detailed architecture of the proposed decoder.

small (undetailed), in extreme poses, or display a wrong identity. Otherwise, the faceset will contain deficient data that impede the training of the model. To remove images that show a false identity or no face at all, we encode all images into the feature space of a pre-trained VGG Network [27], classify them via k-means into $k = 25$ clusters, and remove all clusters that do not contain the person of interest. Faces in extreme poses are identified through their yaw and pitch values, which are computed based on the landmarks obtained previously, while small faces are removed based on their face rectangle size. To identify blurry faces, we compute a blur score based on the variance of the Laplacian of the image and remove all images with a score below a threshold which is identified manually for each faceset. Then, we manually scan the faceset for images that were missed by the previous cleaning steps and delete them. Finally, we remove all images that are too similar to each other using dupeguru [2] and ensure that our faceset contains approximately $4,000$-$8,000$ images, which sharply display the person's face in a variety of poses, expressions and illuminations.

3.2 Model Architecture & Training

We employ a dual-decoder autoencoder architecture, proposed by [6], as our model for deepfake creation. We use the feature extractor of a pre-trained EfficientNet-B4 [28] as the encoder, which is followed by a linear layer, also called the bottleneck. Before the input reaches a decoder, it passes an intermediate block that consists of another linear layer followed by a nearest-neighbor upsampler with LeakyReLU activation. Both decoders in our model share the same architecture, which is inspired by the STOJO model in [6]. We drop their AdaIN block [13], increase the number of residual blocks per upsample layer to two and utilize a sub-pixel upscaler [26]. Accordingly, we obtain a decoder with four upscale layers that can upscale the output of the intermediate block to a spatial resolution of 256^2 pixels. The output of each upscaler passes a LeakyReLu activation. Each upscaler, except the last one, is additionally followed by two residual blocks. The final output of the decoder is computed by a single $2D$ convolutional layer with a sigmoid activation. A visualization of our autoencoder, including a detailed representation of the decoder is given in Figure 1.

Forward-backward pass. A single batch (of size one) consists of two input images, one for each identity. We crop the central 80% of the images and resize them to a spatial resolution of $(256, 256)$ before feeding them to the model. A forward pass of these two inputs results in two output images (one per decoder) of size $(256, 256, 3)$. We train the model such that it learns to reconstruct the faces of each identity. For each input-output pair $(X, \hat{X})$ we compute the loss

$$\begin{aligned}\mathcal{L}(X, \hat{X}) = {} & \mathcal{L}_{Recon}(X * M_{face}, \hat{X} * M_{face}) \\ & + \lambda_{eye} \cdot \mathcal{L}_{Recon}(X * M_{eye}, \hat{X} * M_{eye}) \\ & + \lambda_{mouth} \cdot \mathcal{L}_{Recon}(X * M_{mouth}, \hat{X} * M_{mouth}),\end{aligned}$$

consisting of a reconstruction term, with

$$\mathcal{L}_{Recon}(X, \hat{X}) = \mathcal{L}_{DSSIM}(X, \hat{X}) + \mathcal{L}_{MSE}(X, \hat{X}),$$

where $\mathcal{L}_{DSSIM}$ and $\mathcal{L}_{MSE}$ are given by the DSSIM [31] and MSE metrics respectively. DSSIM utilizes filters which mimic the sensitivity of the human visual system to changes in (high-)frequency components. The combination of DSSIM and MSE enables our model to achieve both accuracy and visual appeal in reconstructing inputs. Moreover, the masks M correspond to the respective facial regions denoted in the subscript. The loss terms concerning the eye and mouth are necessary to punish visible artifacts. In line with [6], we set $\lambda_{eye} = 3$ and $\lambda_{mouth} = 2$. Overall, we are only concerned about the inner part of the face, as we merely want to perform a swap onto another head instead of generating the head entirely. We compute the loss for input-output pairs of both identities at once and then perform an update step on the weights of the model. The forward pass and loss computation are displayed on the left-hand side of Figure 1. This training procedure forces the encoder to learn how to encode an image of any of the two identities into a representation of expression, pose, and illumination that is independent of the identity. Furthermore, the decoders learn to transform this

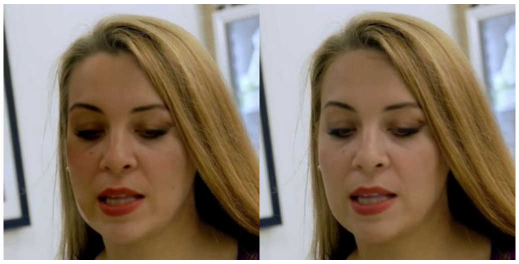

Figure 2: Comparison of the conventional blending procedure and the one proposed by us. Left: Conventional blending with the mask defined by driver. Right: Result with our proposed blending. Best viewed in color.

latent representation into an image of the respective identity with corresponding attributes.

Training details.

- *Optimization.* We use the Adam optimizer [14] with default β and $\epsilon = 1 \times 10^{-7}$. We set the learning rate to 5×10^{-5} and train the model for 1,000,000 steps with a batch size of 32.
- *Data Augmentation.* We apply data augmentation to the input and ground truth images. At first, we perform contrast limited adaptive histogram equalization with a chance of 0.5. Then, the color and lightness parameters in LAB space are randomly adjusted. Moreover, we perform random rotation, scaling, and translation to the color-augmented images. Last, we apply warping to the input images for the first half of training. The warping helps the encoder to generalize its learned representations across identities, while disabling the warping lets the model learn finer details in the end.

3.3 Conversion & Advanced Blending

To swap the face of a target identity onto the head of a driver identity, given an appropriately trained model, we extract the driving face from the frame of interest and align it as described in 3.1. After cropping and resizing, the face image is fed into the model so that the output image is generated by the decoder corresponding to the target identity. This allows us to obtain an image displaying the target with the attributes present in the driver image. Before we re-align the output image and insert it into the driver frame, we apply our advanced blending to merge the output face with the head of the driver. If we want to apply the swap for an entire video, the above procedure is repeated for each frame independently.

Advanced Blending. The conventional blending procedure utilizes Poisson Blending [36] and uses the inner-face segmentation mask of the driver image to indicate where the blending should be performed. However, this can lead to strongly visible artifacts, see Figure 2. Artifacts manifest at the blending boundary on the right edge of the face. Additionally, we observe that the lighting around the right eye is inconsistent with the rest of the face. This is caused by the edge of the blending mask being too close to the edge of either of the faces (driver or fake), hence including parts of the image that lie outside of the face, such as hair or background, in the blending process. To reduce these artifacts, we propose a simple but effective adjustment to the blending mask: If the edge of the blending mask is of suitable distance to the edge of the face, the boundary artifacts disappear. Therefore, we squeeze the mask on each side by 15px and thus increase the distance between the edges of the face and the mask. Note that the squeezing amount is variable and should be adapted to both identities for optimal results, but it is kept constant for all our experiments for consistency. The usage of the squeeze mask effectively removes the boundary artifacts. However, occasional inconsistencies in lighting can still appear when an entire video is manipulated. Hence, we compute the face mask of the generated face as well, squeeze it and intersect it with the squeezed mask of the driver to further exclude regions that lie outside of any of the two faces. The resulting mask is finally used for blending. As shown on the right-hand side of Figure 2, no artifacts appear when our blending procedure is used.

4 EXPERIMENTS

In this section, we present the experiments conducted in order to demonstrate the realism of our fakes as well as the necessity to include high-quality deepfakes in the training of robust detectors for real-world scenarios. We use the "actors" subset of the deepfake detection dataset [9] to train our deepfake autoencoders. The dataset consists of multiple videos of 28 different actors, displaying a variety of poses, expressions, and head movements. We select 11 of the 28 identities and build 13 identity pairings. A faceset for each identity is obtained as described in 3.1. We train a model for each identity pairing and swap the faces in selected videos showing the identities corresponding to the respective model obtaining 90 deepfake videos. We use the corresponding driver videos as the pristine counterpart to our set of forged videos. For simplicity, we refer to the union of these data as "our" dataset. The amount of pristine videos is 77, as the same driver video can be used for multiple forgeries. Some qualitative results of our deepfake generation can be seen in Figure 3.

First, we inspect the performance of a state-of-the-art deepfake detector proposed in [10] on our dataset. Thereafter, we use our data to fine-tune the detector and re-evaluate its performance on hold-out testing sets.

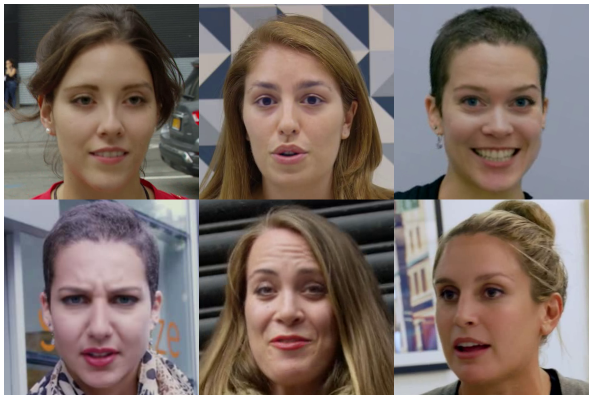

Figure 3: Selected frames of our deepfake dataset. Images generated with six different autoencoders.

Table 1: Test results of RealForensics on our deepfakes and their pristine counterparts from [25]. Testing on our fakes causes a drop in accuracy from 97.4 to 26.7. All scores in %.

AUC	Acc(Pristine)	Acc(Fake)	Acc(Fake+Pristine)
80.2	97.4	26.7	59.3

Table 2: Average AUC scores (in %) of 10 finetuned detectors on the testing subsets of Deepfakes(DF), Faceswap(FS), Face2Face(F2F) and NeuralTextures(NT) in FFPP as well CDF.

DF	FS	F2F	NT	CDF
98.2	95.3	97.7	97.1	78.1

4.1 Testing On High-Quality Fakes

The RealForensics detector [10] learns to classify deepfakes by identifying inconsistencies in facial movement instead of looking for merely simple artifacts. This leads to state-of-the-art generalization performance with respect to unseen datasets and forgery methods. Their approach stands out due to a self-supervised learning framework that utilizes the audio of pristine videos in order to help the model learn stronger representations of facial movement. The detector can classify fakes, even when no audio is available for a given video. We download their fully trained model and prepare our data according to their pre-processing. Then, we test the detector on our dataset. The results are reported in Table 1. They clearly demonstrate that the detector struggles with the detection of high-quality deepfakes. Moreover, the detector's stellar performance on the pristine videos indicates that the poor performance on the fakes is not caused by the shift to another domain of videos. On the one hand, we conclude that our fakes are of sufficient quality to fool detectors that perform well in cross-dataset and cross-manipulation scenarios. On the other hand, we argue that deepfake detectors which are merely trained on research data struggle with well-crafted fakes in real-world scenarios, despite generalizing well across other research datasets.

4.2 Finetuning With High-Quality Fakes

In order to demonstrate the necessity of high-quality fakes in the training sets of deepfake detectors, we fine-tune the RealForensics detector [10] on our dataset. Given the small size of our dataset, we perform a tenfold cross-validation experiment. We separate the dataset into different splits by randomly sampling two exclusive identities for the test-split and one for the validation-split. For a sampled identity, we gather all videos that show their face either as the target in a fake or as the driver in a pristine video and assign them to the corresponding split. We sample 10 train-, validation- and test-splits and ensure that no duplicate splits appear. Thereafter, we fine-tune the fully trained detector on each split and compute performance metrics on the corresponding test-split.

Training details.

- *Optimization.* We use the Adam optimizer [14] with default β and ϵ. We set the learning rate to 8×10^{-5} and train both the backbone and the classification head for 50 epochs with a batch size of 4.
- *Metrics.* We train the model using binary cross-entropy loss, as it is done in [10]. However, we do not use their self-supervised learning approach for the additional training of the backbone. Furthermore, we save the model states with the best performance on the validation-set under the skewed accuracy metric λ_1Acc(Fake) +λ_2Acc(Real). We empirically set $\lambda_1 = 1$ and $\lambda_2 = 3$.

Table 3 shows the accuracy and AUC scores of the fine-tuned detectors on the various test-splits as well as their class distributions. The results clearly indicate that the detector is able to discriminate between our fakes and pristine videos. Hence, we argue that high-quality fakes are able to provide useful information during training. We decided to employ a simple fine-tuning procedure as we aim to demonstrate the utility of the high-quality fakes instead of showing that the detector can perfectly detect our fakes. This can explain the drop in the Acc(Real) metric in Table 3. We hypothesize that sophisticated training with high-quality fakes, a larger training corpus, and, potentially, the self-supervised learning approach by [10] can lead to even better and more robust features for the separation of fake and real faces. Furthermore, to ensure that the detectors' original knowledge is not corrupted by the fine-tuning procedure, we evaluate the fine-tuned models on the testing-sets of FFPP [25] and CDF [18]. The results are displayed in Table 2. We see that the fine-tuned models still perform well on average on FFPP and CDF.

5 CONCLUSION

In this paper, we proposed an autoencoder architecture alongside an extension to the blending procedure for deepfake faceswaps. We thoroughly gathered facesets for 11 different identities and used our model and advanced blending to build a dataset containing 90 high-quality deepfakes. We fed our deepfakes to a state-of-the-art deepfake detector, which was shown to generalize well in cross-dataset and cross-manipulation settings. Thereby, we showed that well-performing detectors trained on common research datasets

Table 3: Test results of the finetuned RealForensics detector on our ten test-splits. All metrics in %. The Avg. metrics are weighted according to the test-split class distributions.

	01	02	03	04	05	06	07	08	09	10	**Avg.**
AUC	83.3	70.5	91.7	95.6	80.3	87.2	88.9	82.2	85.7	92.6	**86.2**
Acc(Real)	68.8	93.4	94.7	100	70.6	61.1	94.4	100	77.8	70.6	**83**
Acc(Fake)	83.3	50	78.9	45.8	78.9	94.1	71.4	55.5	100	94.7	**75.7**
# Samples(Real) : # Samples (Fake)	16:18	16:18	19:19	17:24	17:19	18:17	18:21	15:9	18:21	17:19	**17:19**

still struggle in real-world scenarios. An experiment in which we used our data to finetune a given detector demonstrated that our high-quality fakes possess additional clues for the detection of fakes. These clues highlight the need for more high-quality fakes in the training process of robust detectors and can potentially lead to better generalization results. The latter argument will be the subject of future work.

ACKNOWLEDGMENTS

This work has recieved partial funding by the German Federal Ministry of Education and Research (BMBF) through the Research Program FAKEID under Contract no. 13N15735, as well as the Fraunhofer Society in the Max Planck-Fraunhofer collaboration project NeuroHum.

REFERENCES

[1] Shruti Agarwal and Hany Farid. 2021. Detecting Deep-Fake Videos from Aural and Oral Dynamics. In *2021 IEEE/CVF Conf. on Computer Vision and Pattern Recognition Workshops*. https://doi.org/10.1109/CVPRW53098.2021.00109

[2] arsenetar. 2019. dupeguru. https://github.com/arsenetar/dupeguru

[3] Volker Blanz and Thomas Vetter. 1999. A Morphable Model for the Synthesis of 3D Faces. In *Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '99)*. https://doi.org/10.1145/311535.311556

[4] Adrian Bulat and Georgios Tzimiropoulos. 2017. 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*. https://doi.org/10.1109/ICCV.2017.116

[5] Coin Cheung. 2020. BiSeNet. https://github.com/CoinCheung/BiSeNet

[6] DeepFakes. 2019. faceswap. https://github.com/deepfakes/faceswap

[7] Brian Dolhansky, Joanna Bitton, Ben Pflaum, Jikuo Lu, Russ Howes, Menglin Wang, and Cristian Canton Ferrer. 2020. The DeepFake Detection Challenge Dataset. arXiv:2006.07397 [cs.CV]

[8] Nikita Drobyshev, Jenya Chelishev, Taras Khakhulin, Aleksei Ivakhnenko, Victor Lempitsky, and Egor Zakharov. 2022. MegaPortraits: One-shot Megapixel Neural Head Avatars. In *Proc. of the 30th ACM International Conference on Multimedia*.

[9] Nicholas Dufour, Andrew Gully, Per Karlsson, Alexey Victor Vorbyov, Thomas Leung, Jeremiah Childs, and Christoph Bregler. 2019. DeepFakes Detection Dataset by Google & JigSaw.

[10] Alexandros Haliassos, Rodrigo Mira, Stavros Petridis, and Maja Pantic. 2022. Leveraging Real Talking Faces via Self-Supervision for Robust Forgery Detection. In *Proceedings of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition*.

[11] Alexandros Haliassos, Konstantinos Vougioukas, Stavros Petridis, and Maja Pantic. 2021. Lips Don't Lie: A Generalisable and Robust Approach To Face Forgery Detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 5039–5049.

[12] Karen Hao. 2021. Deepfake porn is ruining women's lives. Now the law may finally ban it. https://www.technologyreview.com/2021/02/12/1018222/deepfake-revenge-porn-coming-ban/

[13] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. 2020. Analyzing and Improving the Image Quality of StyleGAN. In *Proc. CVPR*.

[14] Diederik P. Kingma and Jimmy Ba. 2015. Adam: A Method for Stochastic Optimization. In *3rd International Conference on Learning Representations*.

[15] Pavel Korshunov and Sébastien Marcel. 2018. DeepFakes: a New Threat to Face Recognition? Assessment and Detection. *CoRR* abs/1812.08685 (2018). http://arxiv.org/abs/1812.08685

[16] Lingzhi Li, Jianmin Bao, Ting Zhang, Hao Yang, Dong Chen, Fang Wen, and Baining Guo. 2020. Face X-Ray for More General Face Forgery Detection. In *IEEE/CVF Conf. on Computer Vision and Pattern Recognition*.

[17] Yuezun Li, Ming-Ching Chang, and Siwei Lyu. 2018. In Ictu Oculi: Exposing AI Created Fake Videos by Detecting Eye Blinking. In *2018 IEEE International Workshop on Information Forensics and Security*. https://doi.org/10.1109/WIFS.2018.8630787

[18] Yuezun Li, Pu Sun, Honggang Qi, and Siwei Lyu. 2020. Celeb-DF: A Large-scale Challenging Dataset for DeepFake Forensics. In *IEEE Conference on Computer Vision and Patten Recognition (CVPR)*. Seattle, WA, United States.

[19] Yuchen Luo, Junwei Zhu, Keke He, Wenqing Chu, Ying Tai, Chengjie Wang, and Junchi Yan. 2022. StyleFace: Towards Identity-Disentangled Face Generation on Megapixels. In *Computer Vision – ECCV 2022*.

[20] Falko Matern, Christian Riess, and Marc Stamminger. 2019. Exploiting Visual Artifacts to Expose Deepfakes and Face Manipulations. In *2019 IEEE WACV Workshops*. https://doi.org/10.1109/WACVW.2019.00020

[21] Safa C. Medin, Bernhard Egger, Anoop Cherian, Ye Wang, Joshua B. Tenenbaum, Xiaoming Liu, and Tim K. Marks. 2022. MOST-GAN: 3D Morphable StyleGAN for Disentangled Face Image Manipulation. *Proceedings of the AAAI Conference on Artificial Intelligence* (2022). https://doi.org/10.1609/aaai.v36i2.20091

[22] Yuval Nirkin, Yosi Keller, and Tal Hassner. 2019. FSGAN: Subject agnostic face swapping and reenactment. In *Proceedings of the IEEE ICCV*.

[23] Bo Peng, Wei Xiang, Yue Jiang, Wei Wang, Jing Dong, Zhenan Sun, Zhen Lei, and Siwei Lyu. 2022. DFGC 2022: The Second DeepFake Game Competition. In *2022 IEEE International Joint Conference on Biometrics (IJCB)*. https://doi.org/10.1109/IJCB54206.2022.10007991

[24] Ivan Perov, Daiheng Gao, Nikolay Chervoniy, Kunlin Liu, Sugasa Marangonda, Chris Umé, Mr. Dpfks, Carl Shift Facenheim, Luis RP, Jian Jiang, Sheng Zhang, Pingyu Wu, Bo Zhou, and Weiming Zhang. 2020. DeepFaceLab: A simple, flexible and extensible face swapping framework. *CoRR* abs/2005.05535 (2020). arXiv:2005.05535 https://arxiv.org/abs/2005.05535

[25] Andreas Rössler, Davide Cozzolino, Luisa Verdoliva, Christian Riess, Justus Thies, and Matthias Nießner. 2019. FaceForensics++: Learning to Detect Manipulated Facial Images. In *International Conference on Computer Vision (ICCV)*.

[26] Wenzhe Shi, Jose Caballero, Ferenc Huszar, Johannes Totz, Andrew P. Aitken, Rob Bishop, Daniel Rueckert, and Zehan Wang. 2016. Real-Time Single Image and Video Super-Resolution Using an Efficient Sub-Pixel Convolutional Neural Network. In *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition*.

[27] Karen Simonyan and Andrew Zisserman. 2015. Very Deep Convolutional Networks for Large-Scale Image Recognition. In *3rd International Conference on Learning Representations, ICLR 2015*. http://arxiv.org/abs/1409.1556

[28] Mingxing Tan and Quoc Le. 2019. EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks. In *Proceedings of the 36th International Conference on Machine Learning*. https://proceedings.mlr.press/v97/tan19a.html

[29] Jane Wakefield. 2022. Deepfake presidents used in Russia-Ukraine war. https://www.bbc.com/news/technology-60780142

[30] Yuhan Wang, Xu Chen, Junwei Zhu, Wenqing Chu, Ying Tai, Chengjie Wang, Jilin Li, Yongjian Wu, Feiyue Huang, and Rongrong Ji. 2021. HifiFace: 3D Shape and Semantic Prior Guided High Fidelity Face Swapping. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI-21*. https://doi.org/10.24963/ijcai.2021/157

[31] Zhou Wang, A.C. Bovik, H.R. Sheikh, and E.P. Simoncelli. 2004. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing* 13, 4 (2004), 600–612. https://doi.org/10.1109/TIP.2003.819861

[32] Chao Xu, Jiangning Zhang, Miao Hua, Qian He, Zili Yi, and Yong Liu. 2022. Region-Aware Face Swapping. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 7632–7641.

[33] Yangyang Xu, Bailin Deng, Junle Wang, Yanqing Jing, Jia Pan, and Shengfeng He. 2022. High-resolution face swapping via latent semantics disentanglement. In *Proceedings of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition*.

[34] Xin Yang, Yuezun Li, and Siwei Lyu. 2019. Exposing Deep Fakes Using Inconsistent Head Poses. In *2019 IEEE International Conference on Acoustics, Speech and Signal Processing*. https://doi.org/10.1109/ICASSP.2019.8683164

[35] Changqian Yu, Changxin Gao, Jingbo Wang, Gang Yu, Chunhua Shen, and Nong Sang. 2021. BiSeNet V2: Bilateral Network with Guided Aggregation for Real-Time Semantic Segmentation. *Int. J. Comput. Vision* 129, 11 (nov 2021), 3051–3068. https://doi.org/10.1007/s11263-021-01515-2

[36] Lingzhi Zhang, Tarmily Wen, and Jianbo Shi. 2019. Deep Image Blending. *2020 IEEE Winter Conference on Applications of Computer Vision (WACV)* (2019).

[37] Yuhao Zhu, Qi Li, Jian Wang, Cheng-Zhong Xu, and Zhenan Sun. 2021. One Shot Face Swapping on Megapixels. In *Proceedings of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition*.

[38] Wanyi Zhuang, Qi Chu, Zhentao Tan, Qiankun Liu, Haojie Yuan, Changtao Miao, Zixiang Luo, and Nenghai Yu. 2022. UIA-ViT: Unsupervised Inconsistency-Aware Method Based on Vision Transformer for Face Forgery Detection. In *Computer Vision – ECCV 2022*. Springer Nature Switzerland, Cham, 391–407.

[39] Bojia Zi, Minghao Chang, Jingjing Chen, Xingjun Ma, and Yu-Gang Jiang. 2020. WildDeepfake: A Challenging Real-World Dataset for Deepfake Detection. In *Proceedings of the 28th ACM International Conference on Multimedia*. https://doi.org/10.1145/3394171.3413769