Supplementary Material: CYBORG: Blending Human Saliency Into the Loss Improves Deep Learning-Based Synthetic Face Detection

1. Online Annotation Tool

Fig. 1 shows a screenshot from the tool used to collect the human annotation data, as described in Section 4 of the main paper. An example pair of face images is shown, where the right image is fake. The subject classifying this particular pair of images answered the prompt question correctly by picking the right image as being fake. Green highlights correspond to "salient" regions annotated by the user as to which part(s) of the image led to their classification decision.

2. Datasets

We will release all data collected for this work with the camera-ready submission, along with a standard data sharing license agreement (human annotations were collected under IRB protocol). The data shared with the camera-ready paper will allow for replicating all experiments presented in the paper. Some test sets used in this work are already in the public domain, and their copies can be obtained by following the references provided in the main paper. The data can be used to run the testing codes after performing image pre-processing as described in the main paper.

3. Source Codes

Example source codes demonstrating how CYBORG loss was implemented are available in ***code.zip***. This includes the training and testing code used to compute the results in the paper, along with one example DenseNet121 model that was trained with CYBORG loss. The full suite of pretrained models used for the generation of the paper results will be released with the camera-ready submission. The Xception network code can be downloaded from `https://github.com/ondyari/FaceForensics/tree/master/classification`; the other three architectures can be used natively with PyTorch.

4. Deep Learning-Based Face Segmentation

Examples of the output from the deep learning-based face segmentation tool [11] can be found in Fig. 2. The top row shows three examples of real (genuine) face images as well as their corresponding face segmentations. The bottom rows show three examples of synthetic (fake) face images as well as their corresponding face segmentations. The first two synthetic images shown are from the SREFI dataset [1]) and the third image is generated by StyleGAN2 [6].

5. Supplementary Experimental Results (Sec. 7 in the main paper)

5.1. Tabulated AUCs, Training/Validation Plots, and ROC Curves

Tab. 1 outlines the individual performances of each of the studied approaches on the individual GAN sources; this supplements the plots for all GANs combined, as presented in the main paper.

Figures 3-6 outline the training and validation accuracy during the training processes for all four out-of-the-box architectures studied in this work.

Figures 8-11 show the Receiver Operating Characteristics (ROC) curves for the results from Tab. 1 for all four out-of-the-box architectures. The AUCs from Tab. 1 can be found in the legends of the associated ROC plots.

5.2. Visualization of Model Output CAMs

Detailed here are the model visualizations for all studied architectures. In the manuscript Fig. 7 only ResNet is shown but here all four networks are displayed. This Fig. 7 in this supplementary materials complements Fig. 7 in the paper.

By direct comparison to the average correct human annotation in Fig. 7(e), the models trained with CYBORG focus on features that are more similar to human detailed salient regions than models trained with classical cross-entropy loss in all cases. Thus, it can be concluded these models are effectively guided by the human annotations supplied during the training process.

5.3. Evaluating an Off-the-shelf Deepfake Detector on Test Data

While deepfake technology manipulates real videos by inter-splicing real identities, GAN-generated images are entirely synthetic. Given the slight difference between the two, we wanted to inspect whether or not existing *deepfake* detection methods can be applied to our task of *synthetic image* detection. To verify, we ran a state-of-the-art deepfake detector [2] on our test set of synthetic images.

Prior to evaluating the deepfake models on our test images, model use and accuracy were validated for the ten pretrained models on their respective test sets: DFDC [4] and FFPP [7]. Figure 12 shows the results from the authors' self-reported best ensemble deepfake detection method. On *their* own DFDC and FF++ deepfake test data, the [2] ensemble methods show AUCs of 0.95789 and 0.92047, respectively.

However, when applied in the different domain of synthetic image detection (on *our* synthetic test data), the ensembles of pretrained models are not able to adequately distinguish between real and entirely synthetic images, as seen in Figure 13; the DFDC ensemble method shows an AUC of 0.373 while the FFPP ensemble method shows an AUC of 0.385. These results support the claim that deepfake detection models and synthetic image detection models are ***not*** interchangeable.

5.4. Incorporation Of CYBORG Into An Existing Synthetic Face Detector

In [8], Wang *et al.* design a synthesizer-agnostic classifier of CNN-generated images. To test model generalizability on novel synthesizers, Wang *et al.* trained their model exclusively on ProGAN-generated images. They then tested their model on "never-before-seen" GAN-generated images from StyleGAN [5], CycleGAN [10], and other state-of-the-art methods.

To determine whether the incorporation of CYBORG loss would improve upon this existing method, we conducted experiments under the training scenarios and testing protocols described in the main paper (Sec. 6.1), adding CYBORG loss to Wang *et al.*'s publicly available, re-trainable model [9].

Results (based on the same testing scenarios) can be found in the "CNNDetection" row in Tab. 1, with ROCs for the individual GAN sources in Fig. 14. As can be seen, there is an increase in performance, although relatively minimal compared to that of other models.

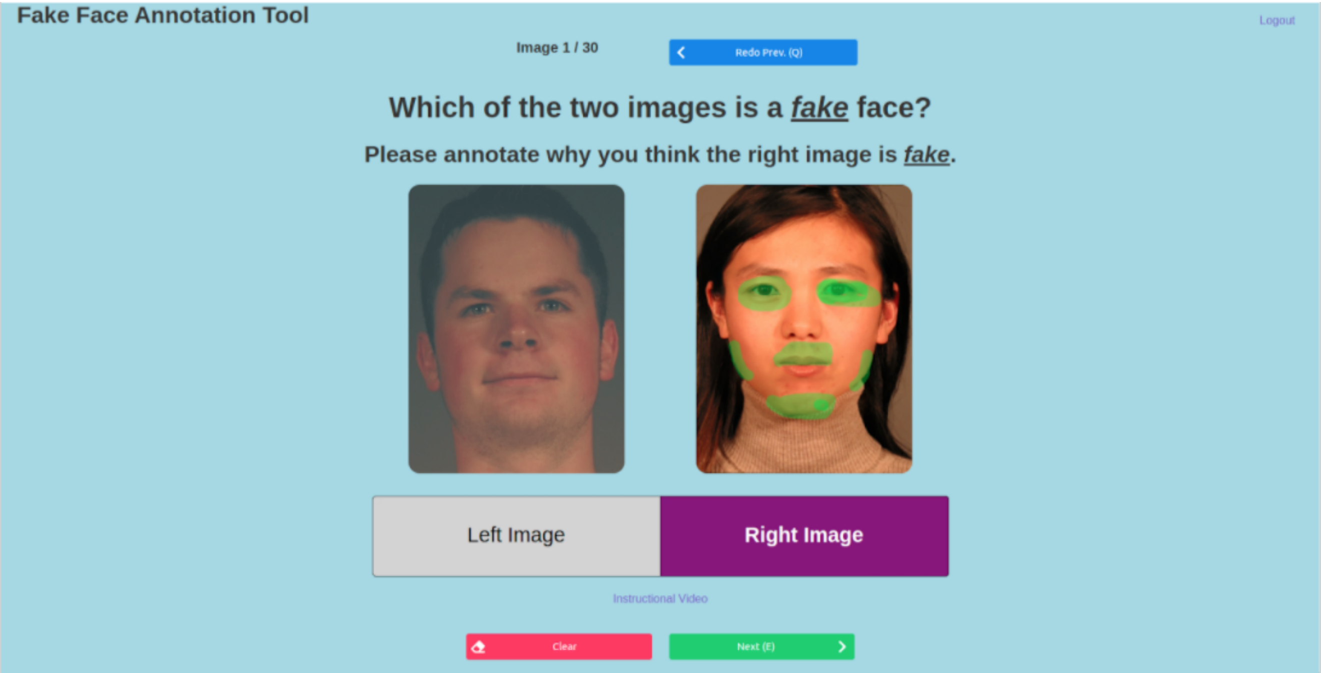
5.5. Incorporation Of Human Saliency Into An Attention Mechanism.

A popular approach to force networks to focus on specified regions is self-attention. In [3], the authors propose a method of self-attention to give extra context to deepfake detection models via image masks. The masks in this approach signal broad areas of "tampered" information, instructing the model where the image alteration has been performed during training. However, attention maps are much coarser as they only provide binary information (tampered / not tampered). Additionally, attention maps are not based on human judgment, but are instead bona-fide ground truth maps of modified regions within an image.

Although not the main goal of this work, we investigate whether the replacement of the masks proposed in [3] with our human saliency maps results in higher accuracy. Because our work focuses on detecting synthetically generated images as a whole, rather than tampered images, real image masks are all zeros and synthetically-generated image masks are all ones, as described in [3]. We train two models in this case: (1) using the original approach with no human saliency, and (2) using our human saliency maps as the masks for real and synthetic images. In both cases, the parameters proposed by the authors are used. The goal of this evaluation is to see whether addition of a self-attention module and human saliency information can also provide a boost in generalization. The "Self-Attention" rows in Tab. 1 illustrate that the replacement of the ground truth masks used in [3] with our human saliency maps increased performance. ROCs on individual GAN sources can also be found in Fig. 15. Since GAN-generated images are entirely synthetic, ground truth data regarding synthetic *regions* within an image typically does not exist. Tab. 1 suggests that implanting human saliency maps into the self-attention module narrows the model's search for areas of importance (even in the absence of ground truth) and boosts performance.

Acknowledgments

Research presented in this paper was supported by the U.S. Department of Defense (Contract No. W52P1J2093009). The views and conclusions contained in this paper are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the U.S. Department of Defense or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes, notwithstanding any copyright notation here on.



Fake Face Annotation Tool

Image 1 / 30

Which of the two images is a ***fake*** face?

Please annotate why you think the right image is ***fake***.

Figure 1: A screenshot from the online annotation tool designed for this work and used to collect human annotation data.

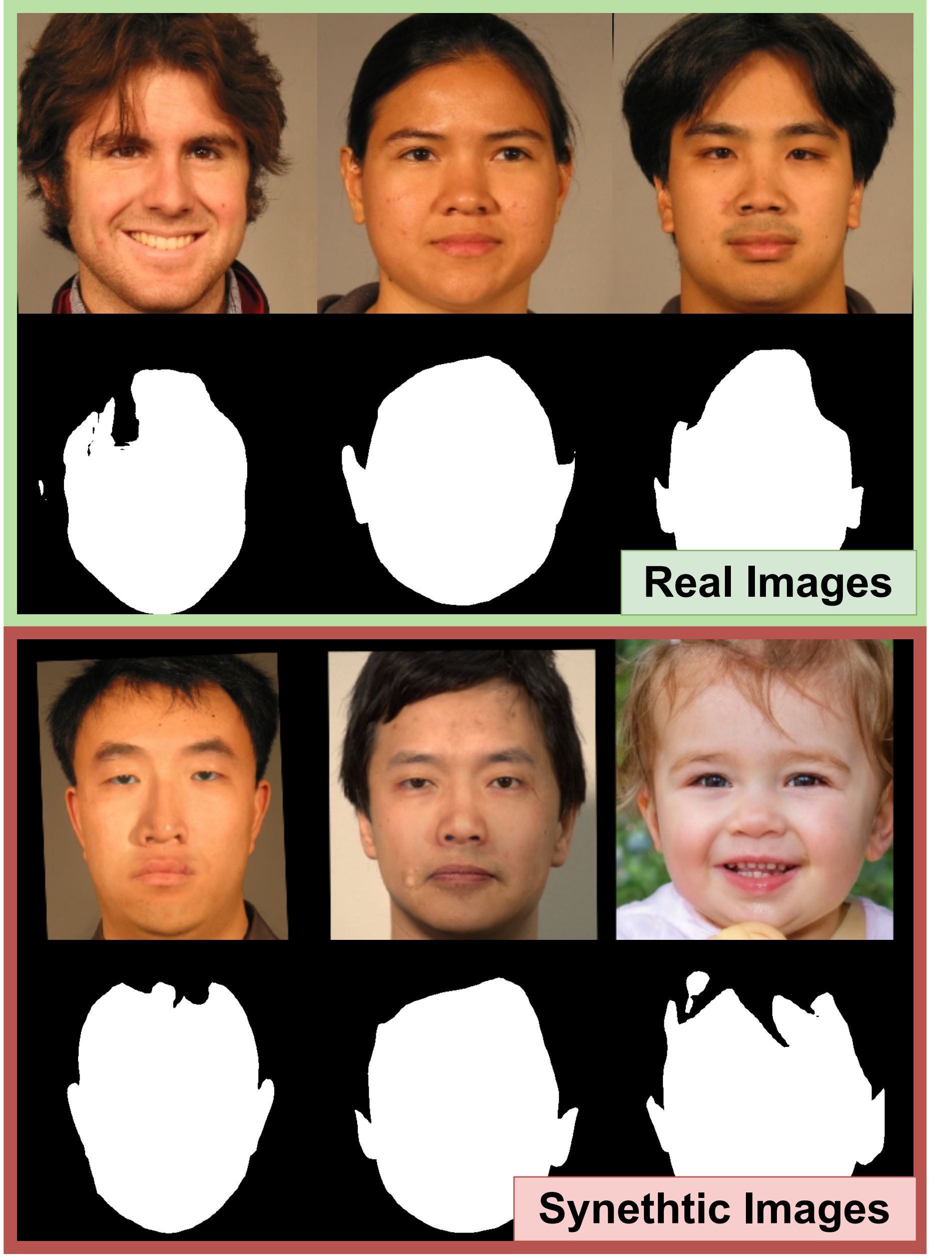


Figure 2: Three examples of real images and corresponding deep learning-based segmentations (top two rows) and three examples of synthetic images and corresponding deep learning-based segmentations (bottom two rows). Bottom two rows: image 1 and 2 from SREFI dataset, image 3 generated by StyleGAN2.

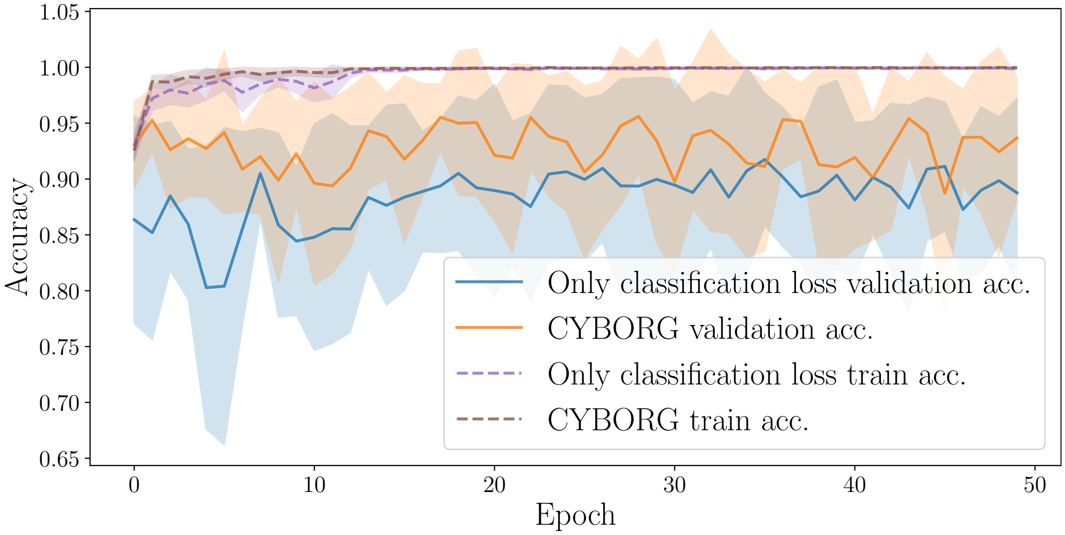


Figure 3: Training / validation accuracy: **DenseNet121**. "Only classification loss" corresponds to Scenario 1 ("Classical Training"), and "CYBORG" corresponds to Scenario 3, as defined in Sec. 6.1 in the main paper.

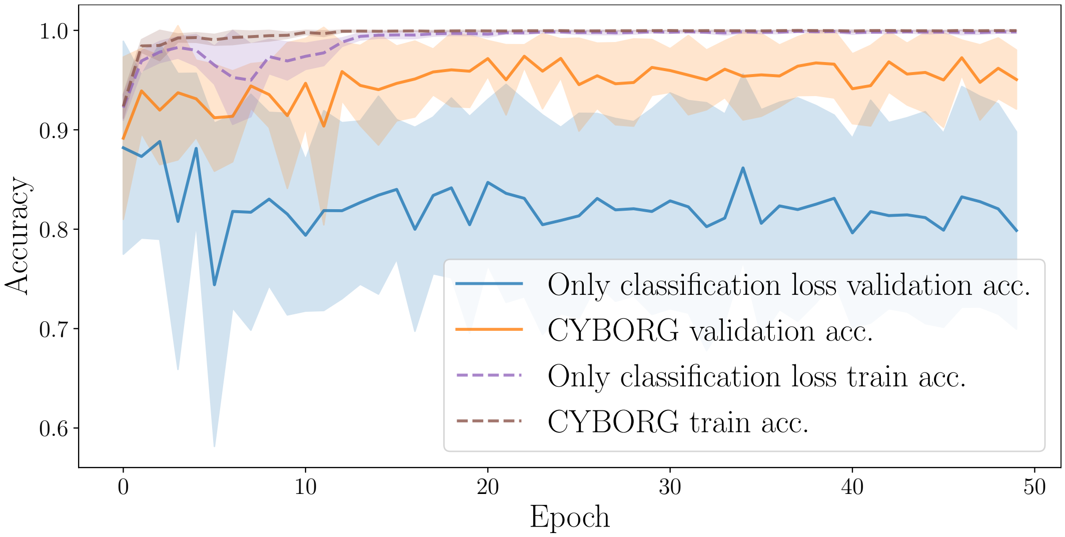


Figure 4: Same as in Fig. 3, but for **ResNet50**

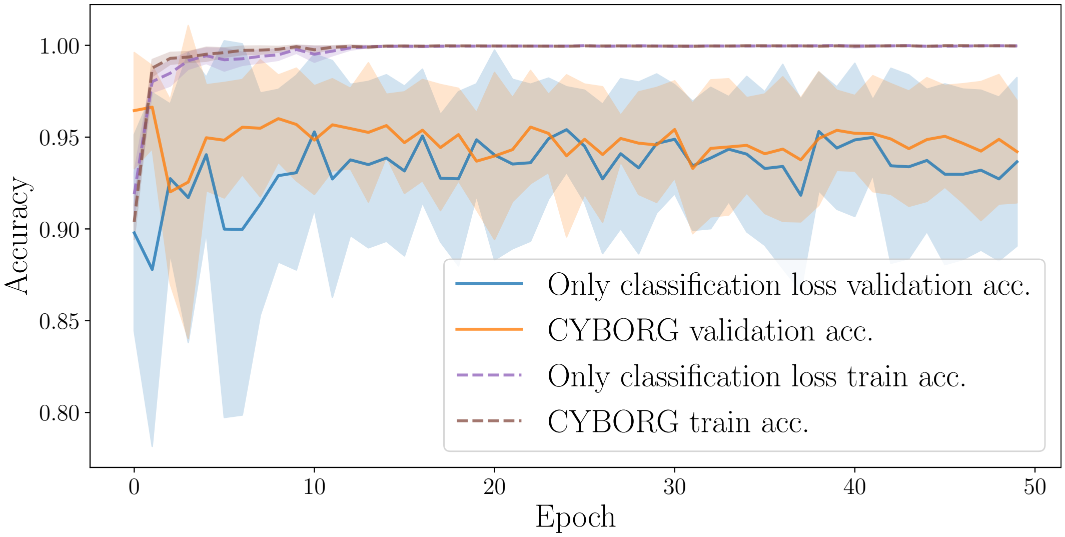


Figure 5: Same as in Fig. 3, but for **Inception-v3**

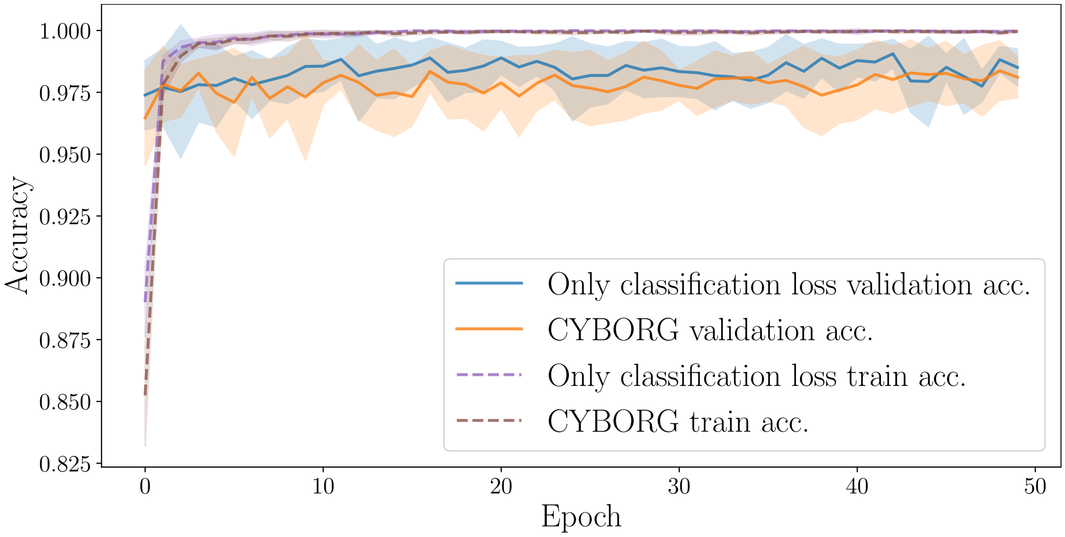
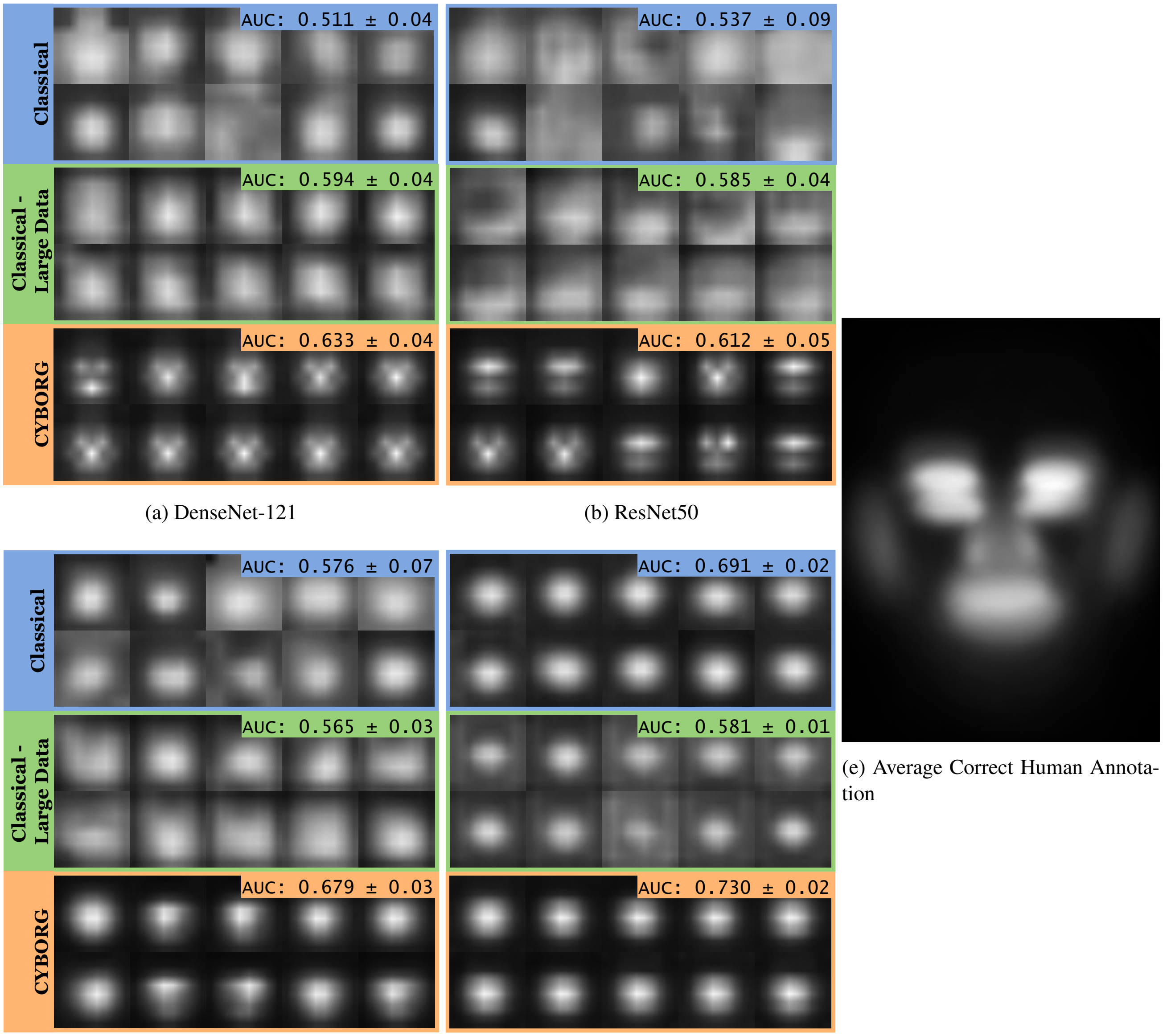


Figure 6: Same as in Fig. 3, but for **Xception Net**



(a) DenseNet-121

(b) ResNet50

(c) Inception-v3

(d) Xception

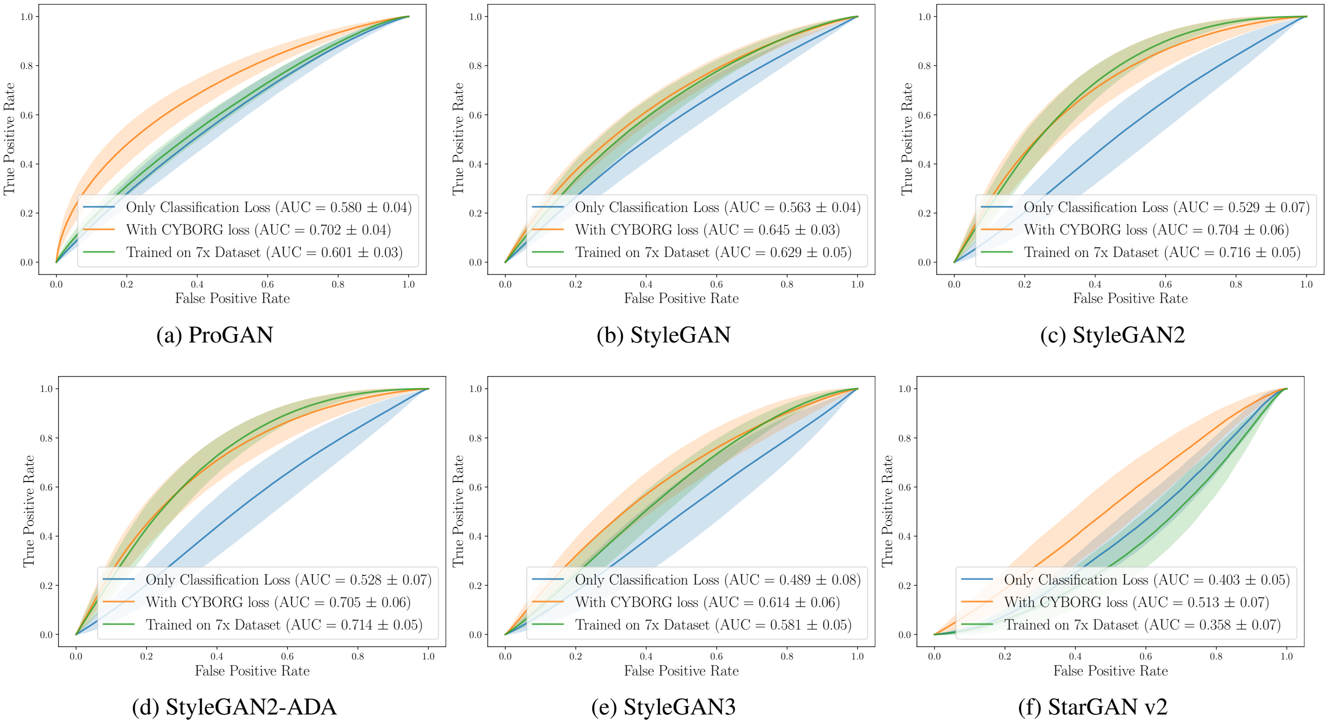
(e) Average Correct Human Annotation

Figure 7: Average CAMs across the entire test set for 10 independently trained models in three experimental settings and four different architectures.

Table 1: Area Under Curve (AUC) with ± one standard deviation (over 10 independent runs of the training-validation experiments) for all combinations of classification models (rows), synthetic face generative models (table section headers) and training strategies (Scenario 1: Classical with cross-entropy loss only; Scenario 2: Classical with Large Data and cross-entropy loss only; Scenario 3: CYBORG – the proposed approach penalizing both the divergence of the model from human saliency and classification performance). Best average AUCs among all scenarios are bold and color-coded: **blue for Classical (Scenario 1)**, **green for Classical with Large Data (Scenario 2)**, and **orange for CYBORG (Scenario 3)**. We can see that in majority of cases, the CYBORG approach results in higher AUCs in a task of recognition of synthetic faces generated by unknown GAN models.

	ProGAN			**StyleGAN**			**StyleGAN2**		
	Classical	CYBORG	Classical – Large Data	Classical	CYBORG	Classical – Large Data	Classical	CYBORG	Classical – Large Data
DenseNet121	0.580 ± 0.04	**0.702 ± 0.04**	0.601 ± 0.03	0.563 ± 0.04	**0.645 ± 0.03**	0.629 ± 0.05	0.529 ± 0.07	0.704 ± 0.06	**0.716 ± 0.05**
ResNet50	0.554 ± 0.06	**0.668 ± 0.04**	0.599 ± 0.03	0.561 ± 0.09	**0.629 ± 0.04**	0.611 ± 0.05	0.556 ± 0.14	0.664 ± 0.06	**0.690 ± 0.07**
Inception v3	0.595 ± 0.07	**0.785 ± 0.04**	0.575 ± 0.03	0.604 ± 0.05	**0.692 ± 0.05**	0.584 ± 0.04	0.630 ± 0.08	**0.801 ± 0.06**	0.697 ± 0.05
Xception Net	**0.740 ± 0.03**	0.725 ± 0.02	0.616 ± 0.02	0.704 ± 0.02	**0.754 ± 0.02**	0.586 ± 0.02	0.826 ± 0.03	**0.873 ± 0.03**	0.710 ± 0.02
CNN Det.	0.521 ± 0.04	**0.525 ± 0.03**	0.461 ± 0.04	0.583 ± 0.02	**0.618 ± 0.03**	0.568 ± 0.04	0.580 ± 0.05	**0.634 ± 0.06**	0.570 ± 0.06
Self-Attention Deepfake Det.	0.483 ± 0.02	**0.489 ± 0.04**	0.500 ± 0.03	0.483 ± 0.02	**0.531 ± 0.04**	0.559 ± 0.04	0.377 ± 0.07	**0.519 ± 0.09**	0.521 ± 0.10

	StyleGAN2-ADA			**StyleGAN3**			**StarGANv2**		
	Classical	CYBORG	Classical – Large Data	Classical	CYBORG	Classical – Large Data	Classical	CYBORG	Classical – Large Data
DenseNet121	0.528 ± 0.07	**0.705 ± 0.06**	0.714 ± 0.05	0.489 ± 0.08	**0.614 ± 0.06**	0.581 ± 0.05	0.403 ± 0.05	**0.513 ± 0.07**	0.358 ± 0.07
ResNet50	0.552 ± 0.14	0.665 ± 0.07	**0.681 ± 0.06**	0.520 ± 0.12	**0.594 ± 0.07**	0.573 ± 0.06	**0.520 ± 0.10**	0.511 ± 0.07	0.372 ± 0.07
Inception v3	0.631 ± 0.08	**0.808 ± 0.05**	0.697 ± 0.06	0.557 ± 0.07	**0.701 ± 0.10**	0.546 ± 0.04	**0.468 ± 0.13**	**0.468 ± 0.07**	0.305 ± 0.05
Xception Net	0.818 ± 0.03	**0.868 ± 0.03**	0.699 ± 0.03	0.701 ± 0.03	**0.771 ± 0.03**	0.500 ± 0.03	0.431 ± 0.05	**0.473 ± 0.02**	0.366 ± 0.04
CNN Det.	0.576 ± 0.05	**0.632 ± 0.06**	0.566 ± 0.06	0.523 ± 0.04	**0.578 ± 0.06**	0.522 ± 0.06	0.525 ± 0.06	**0.555 ± 0.03**	0.552 ± 0.08
Self-Attention Deepfake Det.	0.376 ± 0.07	0.518 ± 0.09	**0.521 ± 0.10**	0.409 ± 0.06	**0.522 ± 0.09**	0.516 ± 0.10	**0.434 ± 0.04**	0.418 ± 0.07	0.388 ± 0.08



(a) ProGAN

(b) StyleGAN

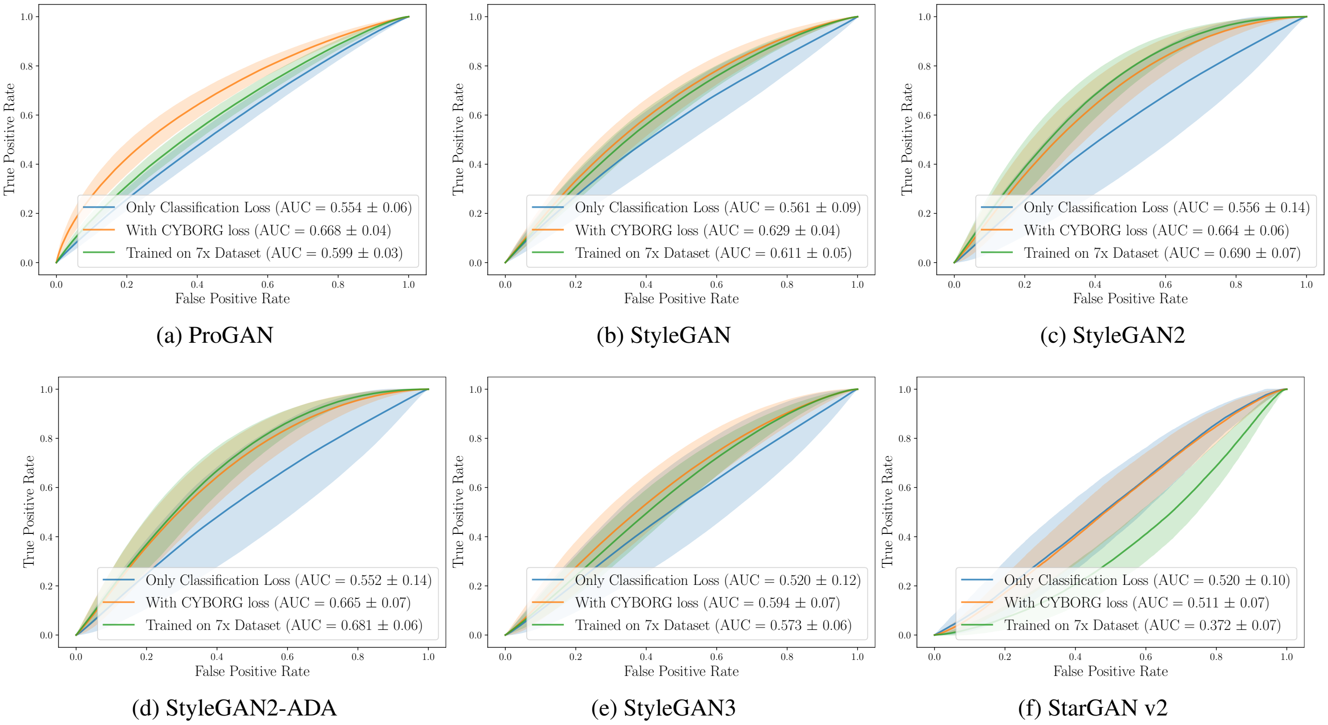
(c) StyleGAN2

(d) StyleGAN2-ADA

(e) StyleGAN3

(f) StarGAN v2

Figure 8: ROC curves associated with results shown in Tab. 1. Classification model: **DenseNet121**. "Only classification loss" corresponds to Scenario 1 ("Classical Training"), "Trained with 7x Dataset" corresponds to Scenario 2 ("Classical – Large Data"), and "CYBORG" corresponds to Scenario 3, as defined in Sec. 6.1 in the main paper.



(a) ProGAN

(b) StyleGAN

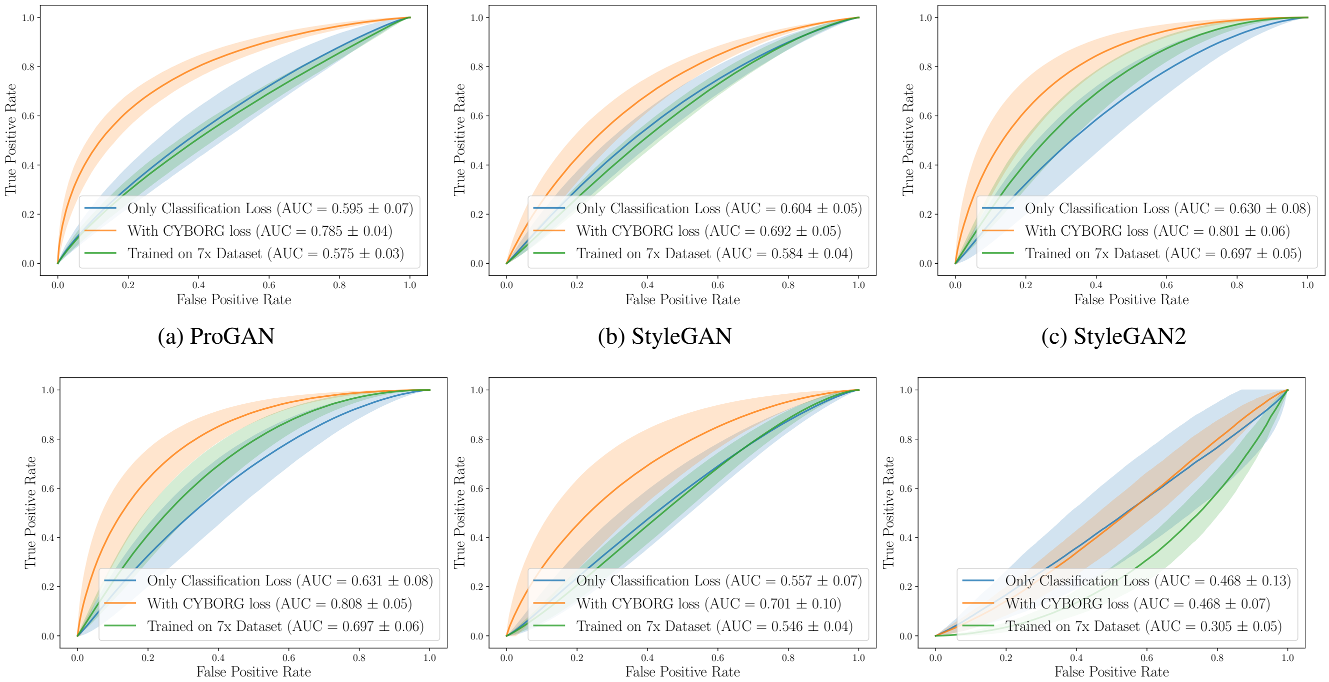
(c) StyleGAN2

(d) StyleGAN2-ADA

(e) StyleGAN3

(f) StarGAN v2

Figure 9: Same as in Fig. 8, except for classification model: **ResNet50**



(a) ProGAN

(b) StyleGAN

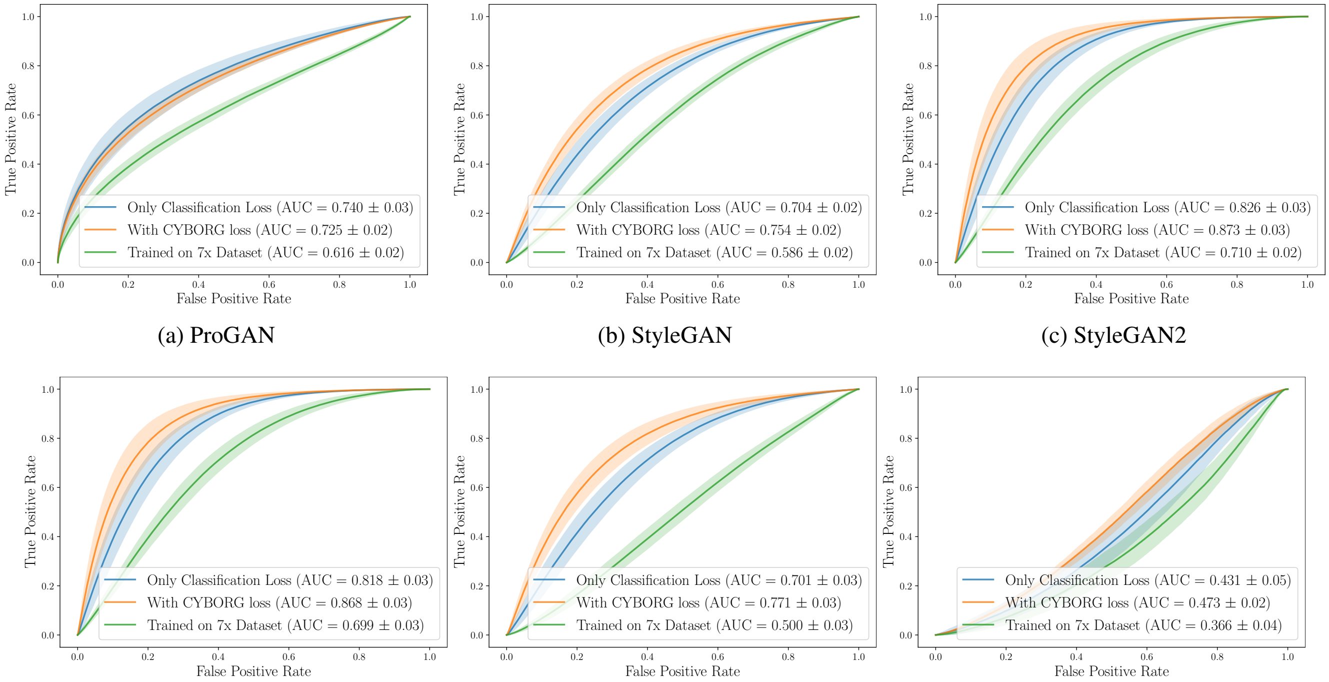
(c) StyleGAN2

(d) StyleGAN2-ADA

(e) StyleGAN3

(f) StarGAN v2

Figure 10: Same as in Fig. 8, except for classification model: **Inception-v3**



(a) ProGAN

(b) StyleGAN

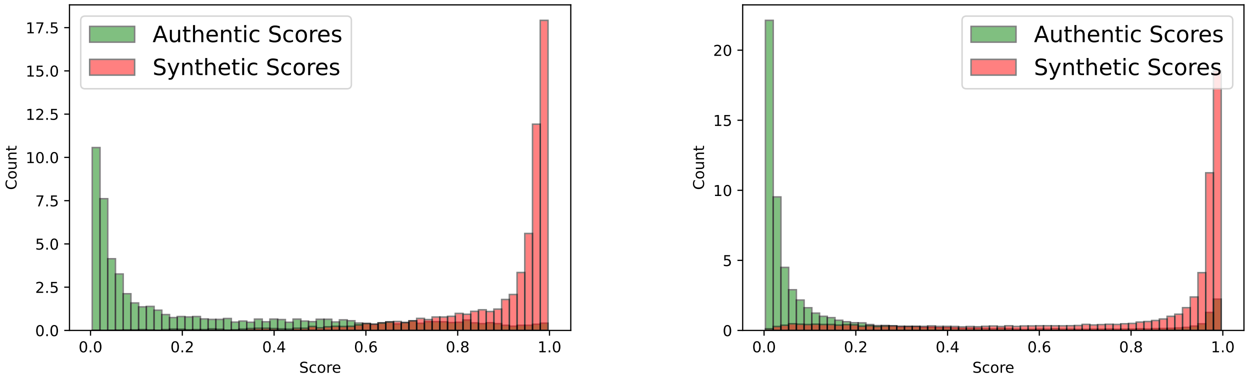
(c) StyleGAN2

(d) StyleGAN2-ADA

(e) StyleGAN3

(f) StarGAN v2

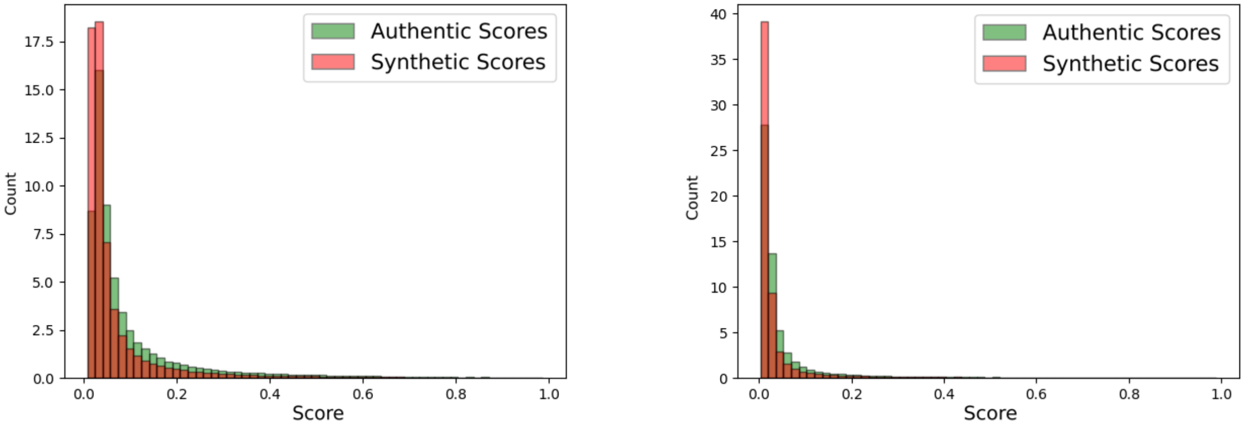
Figure 11: Same as in Fig. 8, except for classification model: **Xception Net**



(a) DFDC

(b) FF++

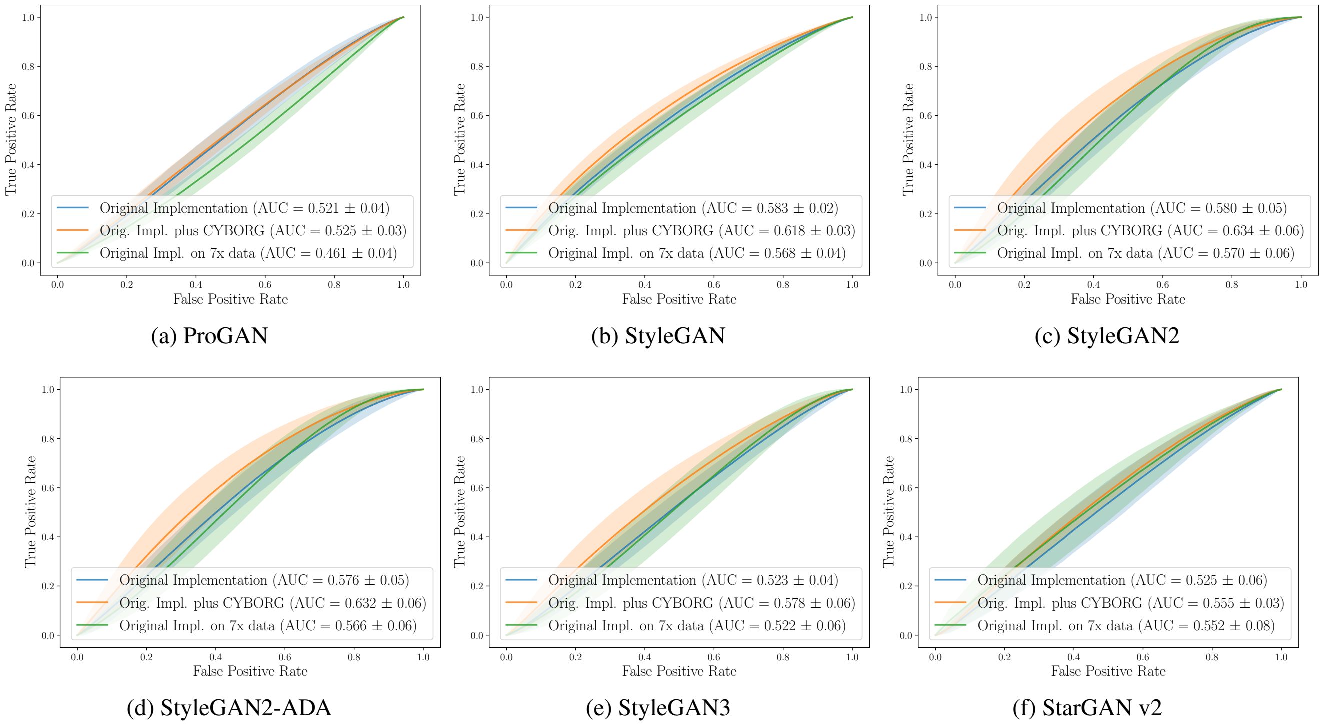
Figure 12: Best reported ensemble method among off-the-shelf deepfake detector models when applied to **the original authors'** respective DFDC and FF++ deepfake test data. As in the original paper, this deepfake detector performs very well on deepfake samples.



(a) DFDC

(b) FF++

Figure 13: Best reported ensemble method among off-the-shelf deepfake detector models when applied to **our synthetic face test data**. As it can be seen, the method designed for deep fakes detection is not able to detect synthetically-generated faces.



(a) ProGAN

(b) StyleGAN

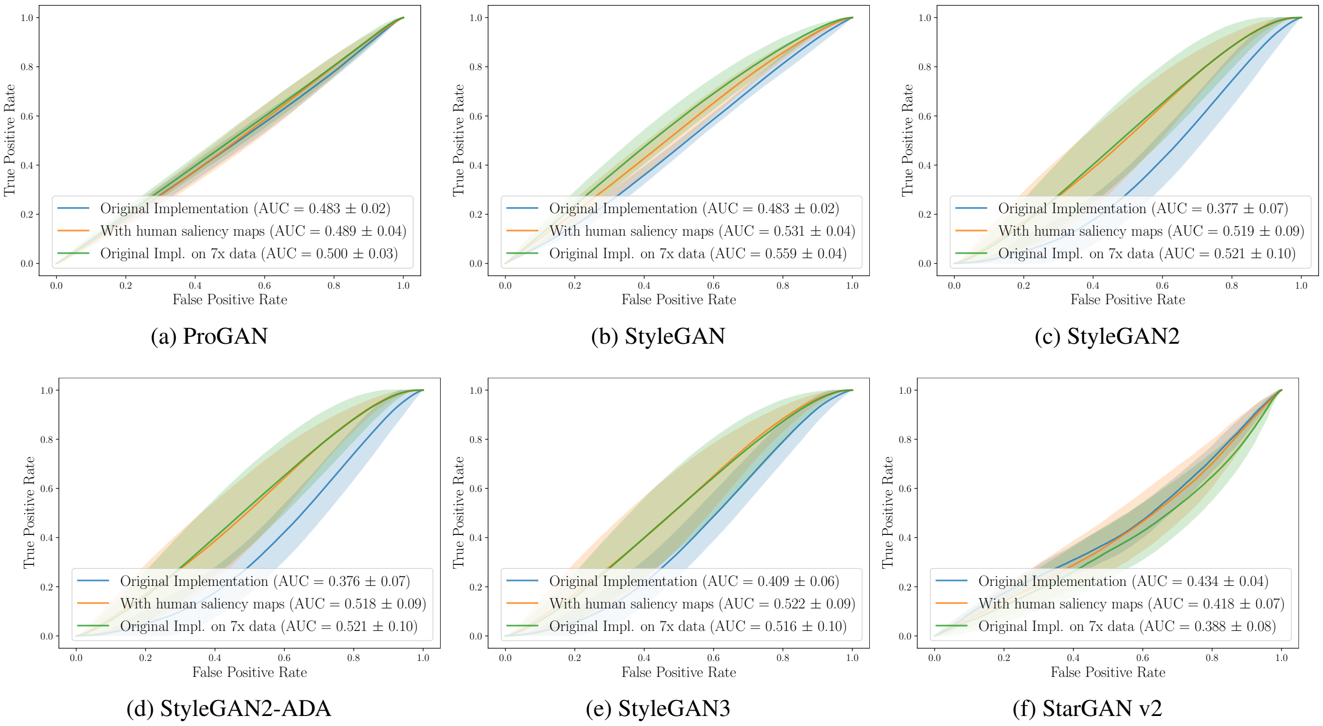
(c) StyleGAN2

(d) StyleGAN2-ADA

(e) StyleGAN3

(f) StarGAN v2

Figure 14: Same as in Fig. 8, except for classification model: **CNNDetection**



(a) ProGAN

(b) StyleGAN

(c) StyleGAN2

(d) StyleGAN2-ADA

(e) StyleGAN3

(f) StarGAN v2

Figure 15: Same as in Fig. 8, except for classification model: **Self-attention**

References

[1] Sandipan Banerjee, John S. Bernhard, Walter J. Scheirer, Kevin W. Bowyer, and Patrick J. Flynn. Srefi: Synthesis of realistic example face images. In *IEEE Int. Joint Conf.on Biometrics (IJCB)*, pages 37–45, 2017.

[2] Nicolo Bonettini, Edoardo Daniele Cannas, Sara Mandelli, Luca Bondi, Paolo Bestagini, and Stefano Tubaro. Video face manipulation detection through ensemble of cnns. In *2020 25th International Conference on Pattern Recognition (ICPR)*, pages 5012–5019. IEEE, 2021.

[3] Hao Dang, Feng Liu, Joel Stehouwer, Xiaoming Liu, and Anil K Jain. On the detection of digital face manipulation. In *IEEE/CVF Conf. Comput. Vis. Pattern Recog. (CVPR)*, pages 5781–5790, 2020.

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

[5] Tero Karras, Samuli Laine, and Timo Aila. A Style-Based Generator Architecture for Generative Adversarial Networks. In *IEEE/CVF Conf. Comput. Vis. Pattern Recog. (CVPR)*, pages 4401–4410, 2019.

[6] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and Improving the Image Quality of StyleGAN. In *IEEE/CVF Conf. Comput. Vis. Pattern Recog. (CVPR)*, pages 8110–8119, 2020.

[7] Andreas Rössler, Davide Cozzolino, Luisa Verdoliva, Christian Riess, Justus Thies, and Matthias Nießner. FaceForensics++: Learning to detect manipulated facial images. In *International Conference on Computer Vision (ICCV)*, 2019.

[8] Sheng-Yu Wang, Oliver Wang, Richard Zhang, Andrew Owens, and Alexei A Efros. Cnn-generated images are surprisingly easy to spot ... for now. In *IEEE/CVF Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2020.

[9] Sheng-Yu Wang, Oliver Wang, Richard Zhang, Andrew Owens, and Alexei A Efros. CNNDetection. `https://github.com/peterwang512/CNNDetection`, 2020.

[10] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, 2017.

[11] zll. BisSeNet – Face Parsing Tool. `https://github.com/zllrunning/face-parsing.PyTorch`, 2019.