Safe Fakes: Evaluating Face Anonymizers for Face Detectors Supplementary Material

Sander R. Klomp[12], Matthew van Rijn[3], Rob G.J. Wijnhoven[2],
Cees G.M. Snoek[3], Peter H.N. de With[1]
[1] Department of Electrical Engineering, Eindhoven University of Technology, Eindhoven, The Netherlands
[2] ViNotion B.V., Eindhoven, The Netherlands
[3] Informatics Institute, University of Amsterdam, Amsterdam, The Netherlands

I. Introduction

This is the supplementary material of the paper 'Safe Fakes: Evaluating Face Anonymizers for Face Detectors'. We provide additional information and experiments to extend the main paper and to confirm the generalization of its conclusions.

- We provide additional information on the face sizes in the easy/medium/hard split of the WIDER FACE dataset [6].
- The main paper shows graphs only for results obtained using the Dual-Shot Face Detector [2] (DSFD), while this supplementary material also contains results of the same experiments using different detectors.
- Some additional examples of anonymized images are given to provide more feeling for the behavior of each anonymization method.
- We show the effect of face detector training iterations separated per difficulty instead of averaged as in the main paper.

II. Wider face difficulty split size distribution

In the main paper we discuss results of the DSFD detector on three splits of the WIDER FACE validation set: 'easy', 'medium' and 'hard'. These splits are defined in the WIDER FACE validation dataset, as how well the faces could be detected using the EdgeBox detector. For more details on this process, we refer to the original WIDER FACE paper [6]. These difficulty levels correlate with the amount of blur, occlusion, strange poses and uncommon illumination, all properties that are labeled for each face the WIDER FACE dataset. They likely also correlate with size, although [6] do not explicitly show this. We are specifically interested in the sizes of faces for several reasons. First of all, for the anonymization using GANs, the size of the face determines whether keypoints can be extracted and thus whether a realistic face can be generated. As mentioned in the main paper, for CIAGAN [3] and Sun2018 [5] the minimum face size is around 50 pixels, while for DeepPrivacy [1] it is around 14 pixels. Face size is defined as the square root of the area of each ground truth face bounding box. Second, it is easier for a GAN to generate small faces than large faces, because higher resolution faces require more detail to be 'hallucinated' by the GAN. Third, the face size determines how severe the 'FullBlur' method anonymization blurs the face. FullBlur blurs the entire image with a single blur kernel of fixed size (18×18 pixels), regardless of face size.

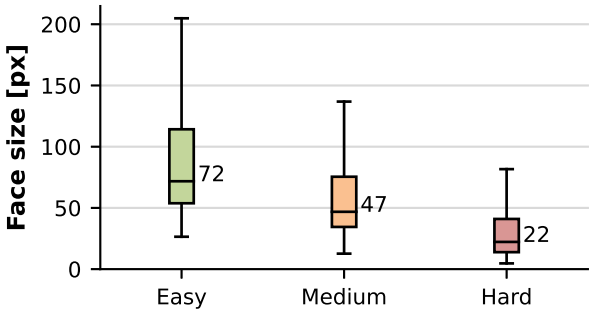

Fig. 1: Boxplot of face size (square root of area) for each difficulty split in the WIDER FACE validation set. Median values are noted next to the boxes. The majority of 'Hard' faces are below the 50 pixel threshold of CIAGAN and Sun2018 and the median 'Hard' face is close in size to the 18 pixel FullBlur kernel.

The distribution of face sizes per difficulty is shown in Fig. 1. The median value of each difficulty is shown next to each box: 72 pixels for 'easy' faces, 47 pixels for 'medium' faces and '22' pixels for hard faces. It is clear that the hard faces are mostly smaller than the 50 pixels required for CIAGAN and Sun2018. Furthermore they are so small that an 18-pixel blur kernel would remove nearly all facial features. An example of the latter is shown in Fig. 2. Clearly, these blurred faces are not suited for training a face detector. In the main paper we discuss the reason for the surprisingly high scores on 'hard' faces even when training with 'FullBlur' training data: scaling data augmentation during detector training. This is shown in the original paper in Fig. 4 and for other detectors in this supplement in Figs. 4 and 5. Although the small faces in the original training dataset are made unusable by blurring with an 18×18 pixel kernel, large faces remain recognizable. Downscaling these large slightly-blurred faces during training still allows the network to learn what small faces look like, without having any recognizable small faces in the training set.

Fig. 2: Example of a crop of several faces of approximately 22 pixels wide (the median size of 'hard' faces) that have been blurred with the 18-pixel 'FullBlur' kernel. All facial features are unrecognizable.

III. Repeated experiments using different detectors

All experiments in the main paper have been performed using the DSFD detector. Although DSFD is a state-of-the-art face detector, we repeat the experiments with the popular Faster-RCNN detector and the current best best-performing face detector on WIDER FACE[1]. To reinforce the general conclusions, the experiments of Figs. 4, 7 and 9 of the main paper that were performed with DSFD [2] (a dual-shot VGG-based architecture) (2019) have been recreated with both the TinaFace face detector [**?**] (2020), a single-shot ResNet-based architecture, and Faster-RCNN [4] (2015), a two-shot ResNet-based general object detection network. These networks allow us to verify the conclusions for a different backbone, single-shot or two-shot architecture and a network that was not specifically designed to detect faces. The text in the following subsections will regularly reference the figures in the main paper.

A. Faster-RCNN

Training parameters We employ the Torchvision implementation with feature pyramid and ImageNet-pretrained ResNet-50 backbone[2]. We add anchors of size 16 to improve detection of small faces (with 0.06 mAP) and set the learning rate to 10^{-3}, the momentum to 0.9, the weight decay to 10^{-4} and decrease the learning rate by a factor of 10 every 3 epochs. We use the same data augmentation as DSFD[3] and train for 10 epochs with a batch size of 4. The first three layers of the ImageNet pretrained backbone are frozen during training. Training for 10 epochs is relatively short, but we noticed very little performance improvement after the first 2-4 epochs regardless of how we tweaked the learning rate and weight decay.

Results and conclusions The equivalent graphs of Figs. 4, 7 and 9 of the main paper are shown in Figs. 4, 7 and 6, respectively. Looking at Figs. 4, the general object detector Faster-RCNN appears to be significantly less suited for the face detection task than the specialized face detector DSFD, scoring an mAP of only 0.34 on the 'hard' images compared to DSFD with 0.86. This low performance approximately matches the scores given in the original WIDER FACE paper, where a generic 'Two-stage CNN' achieved an mAP of 0.30 on 'hard' (Fig. 8c in [6]). Due to the low detection scores the performance degradation of all anonymization methods is smaller compared to DSFD. The overal trends between different anonymization methods remain similar to what we concluded for DSFD: both FullBlur and DeepPrivacy perform very close to original data while all other conventional methods vastly reduce face detection performance. DeepPrivacy remains the best GAN-based method, although the gap with CIAGAN is smaller now that the detector is less powerful (from average 0.28 mAP decrease to 0.05 mAP). Apparently, less accurate detectors suffer less from the anonymization artefacts, which supports the conclusion of Fig. 6 in the main paper. The faces in the long tail of the distribution are not accurately detected and therefore artefacts in these faces do not significantly harm the detector training. This is confirmed by the lack of change when increasing the number of DeepPrivacy training iterations (Fig. 6), and the lack of change when increasing the number of training faces of DeepPrivacy (Fig. 7). Both figures show that the reduction of anonymization artefacts does not benefit the detector.

B. TinaFace

Training parameters We employ the official repository[4] and only change the number of training iterations to 60,000 (from 2M) to match our DSFD training time and train on a single GPU instead of three. This reduces the mAP on 'hard' faces from 0.92 to 0.88. All other training parameters are unchanged. This also means that we now use the data augmentation of TinaFace instead of the one of DSFD.

Results and conclusions The equivalent graphs of Figs. 4, 7 and 9 are shown in Figs. 5, 9 and 8 respectively. With a similar number of training iterations as DSFD, TinaFace also performs very similarly despite being a single-shot instead of dual-shot detector and using a different backbone (ResNet-50 instead of VGG-16). In Fig. 5 the same trends can be seen as for DSFD with the notable observation of the much stronger performance degradation with Blur and FullBlur,

[1] paperswithcode.com/sota/face-detection-on-wider-face-hard

[2] fasterrcnn_resnet50_fpn in `pytorch.org/vision/stable/_modules/torchvision/models/detection/faster_rcnn`

[3] DSFD code available at `github.com/Tencent/FaceDetection-DSFD`

[4] TinaFace code: `github.com/Media-Smart/vedadet`, which also contains other detectors

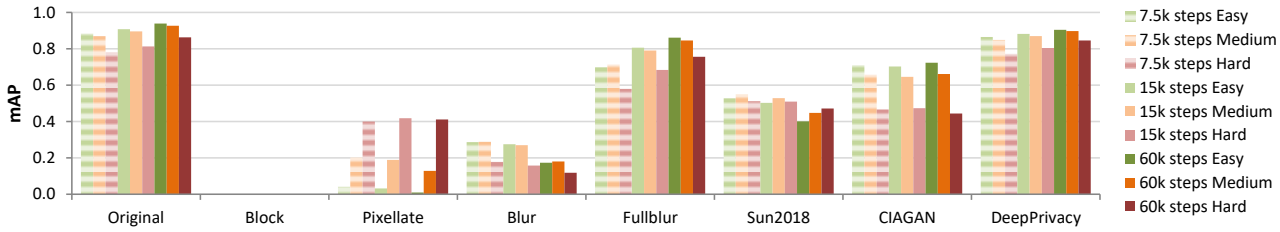

Fig. 3: **What is the influence of face detector quality?** Detector training iterations versus mAP score. This is the same graph as Figure 6 in the main paper, but without averaging over all difficulties.

when compared to the same graph of DSFD (Fig. 4 in the main paper). Interestingly, this is not caused primarily by the detector itself, but by the different data augmentation that TinaFace employs during training. DSFD has very strong scale augmentation: e.g. it can even randomly downscale a relatively common 'easy' face from 200×200 pixels to 16×16 pixels (scale factor 12). In contrast, TinaFace does not explicitly augment scale and only implicitly has scaling augmentation due to random crops of at least 0.3 times the size of the image that get resized to a fixed size during training (scale factor 3). Figs. 9 and 8 show the same trends as for DSFD with the only small difference being that TinaFace performs slightly better (0.01 mAP) on 'medium' than on 'easy' faces, while the opposite is true for DSFD.

IV. Additional anonymization examples

In the main paper, several unexpected properties of the anonymization process are mentioned.

- Blur uses a fixed blur kernel, which also means larger faces are anonymized less well.
- Pixelate transforms the face to a fixed number of pixels, causing it to have limited effect on small faces and a stronger effect on large faces, the exact opposite of blurring.
- FullBlur uses a fixed-size blur kernel regardless of face size, which means large faces are hardly anonymized.
- Sun2018 can generate faces without keypoints, but with significantly reduced quality.
- CIAGAN is completely unable to generate faces without detectable keypoints.
- DeepPrivacy can almost always generate a face, as its small number of keypoints can be detected even on tiny faces. In some cases, such as strange viewpoints, some artefacts are still clearly visible.

We show four image examples with highly varying sizes and types of faces in Fig. 10, additional to the single example in Fig. 1 in the main paper. A remarkable result is shown in the bottom right of Fig. 10, where the skull has been 'anonymized' by DeepPrivacy while not being a ground truth face.

V. Effect of face detector training iterations for each difficulty

Fig. 3 shows an expanded version of Fig. 6 of the main paper which allows more detailed comparisons. The number of bars somewhat obfuscates the general trends, so for the general trends see the original figure. Overall, the same observations from Figs. 4 and 6 can now be made from this single graph, with no unexpected results.

References

[1] H. Hukkelås, R. Mester, and F. Lindseth. Deepprivacy: A generative adversarial network for face anonymization. In *International Symposium on Visual Computing*, pages 565–578. Springer, 2019.

[2] J. Li, Y. Wang, C. Wang, Y. Tai, J. Qian, J. Yang, C. Wang, J. Li, and F. Huang. DSFD: Dual shot face detector. In *CVPR*, 2019.

[3] M. Maximov, I. Elezi, and L. Leal-Taixé. CIAGAN: Conditional Identity Anonymization Generative Adversarial Networks. In *CVPR*, 2020.

[4] S. Ren, K. He, R. Girshick, and J. Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. *NIPS*, pages 91–99, 2015.

[5] Q. Sun, L. Ma, S. Joon Oh, L. Van Gool, B. Schiele, and M. Fritz. Natural and effective obfuscation by head inpainting. In *CVPR*, 2018.

[6] S. Yang, P. Luo, C. C. Loy, and X. Tang. WIDER FACE: A Face Detection Benchmark. In *CVPR*, 2016.

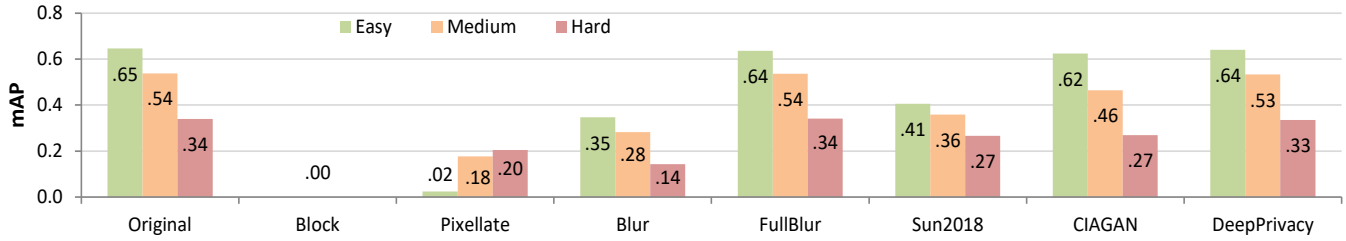

Fig. 4: **Which face anonymizer performs best?** Faster-RCNN mAP scores on WIDER FACE validation set, per difficulty after being trained on the anonymized training set using different methods. DeepPrivacy, FullBlur and CIAGAN all perform similarly and close to original, but scores overall are much lower than for DSFD and TinaFace.

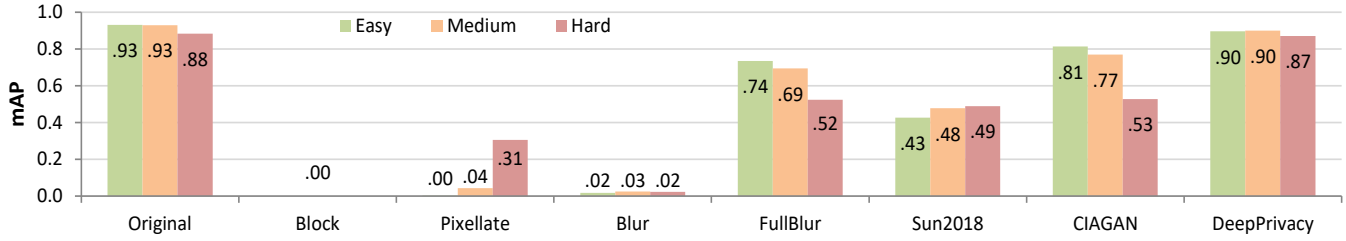

Fig. 5: **Which face anonymizer performs best?** TinaFace mAP scores on WIDER FACE validation set, per difficulty after being trained on the anonymized training set using different methods. DeepPrivacy performs best, with only a 1-3% mAP score drop depending on face difficulty, same as for DSFD.

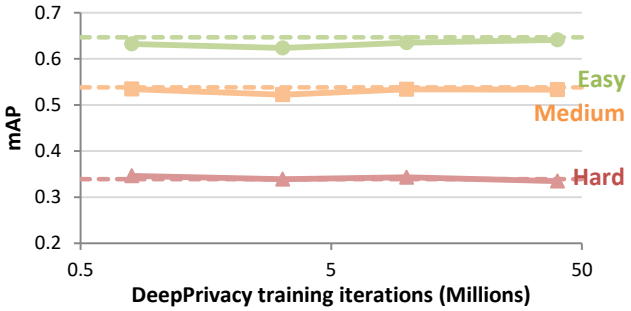

Fig. 6: **Why DeepPrivacy performs best?** mAP scores of Faster-RCNN trained on data anonymized by DeepPrivacy as a function of the number of iterations. Dashed lines show scores of original data. Default Faster-RCNN does not reach high enough performance for the small differences in generated anonymized face quality to matter and just always matches training with original data.

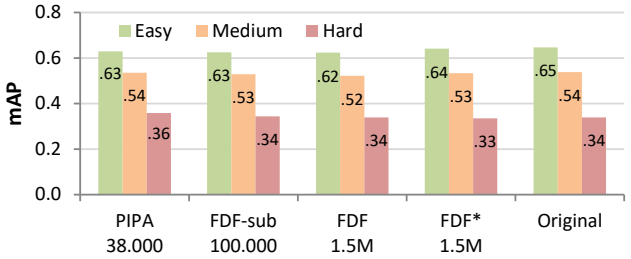

Fig. 7: **Why DeepPrivacy performs best?** Faster-RCNN performance for increasing DeepPrivacy training dataset size. FDF* shows DeepPrivacy trained for the full 40M (vs 4M) iterations. Default Faster-RCNN does not reach high enough performance for the small differences in generated anonymized face quality to matter.

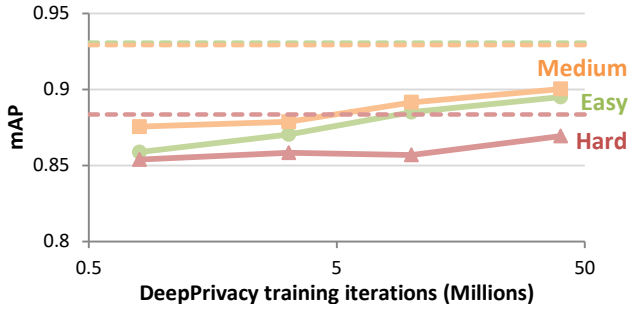

Fig. 8: **Why DeepPrivacy performs best?** mAP scores of TinaFace trained on data anonymized by DeepPrivacy as a function of the number of iterations. Dashed lines show scores of original data. Training the anonymizer for more iterations keeps improving detector performance, with no clear sign of a plateau, even after millions of iterations. This is the same conclusion as for this experiment using DSFD.

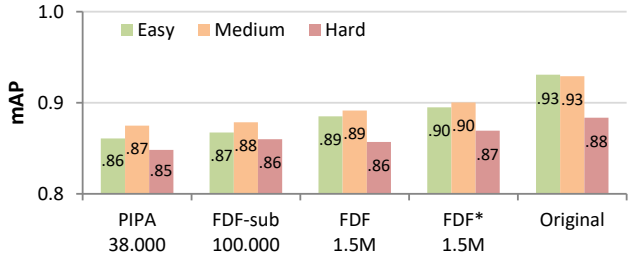

Fig. 9: **Why DeepPrivacy performs best?** TinaFace performance for increasing DeepPrivacy training dataset size. FDF* shows DeepPrivacy trained for the full 40M (vs 4M) iterations. Training the anonymizer with a larger dataset improves results, with a similar trend as the DSFD results in the main paper.

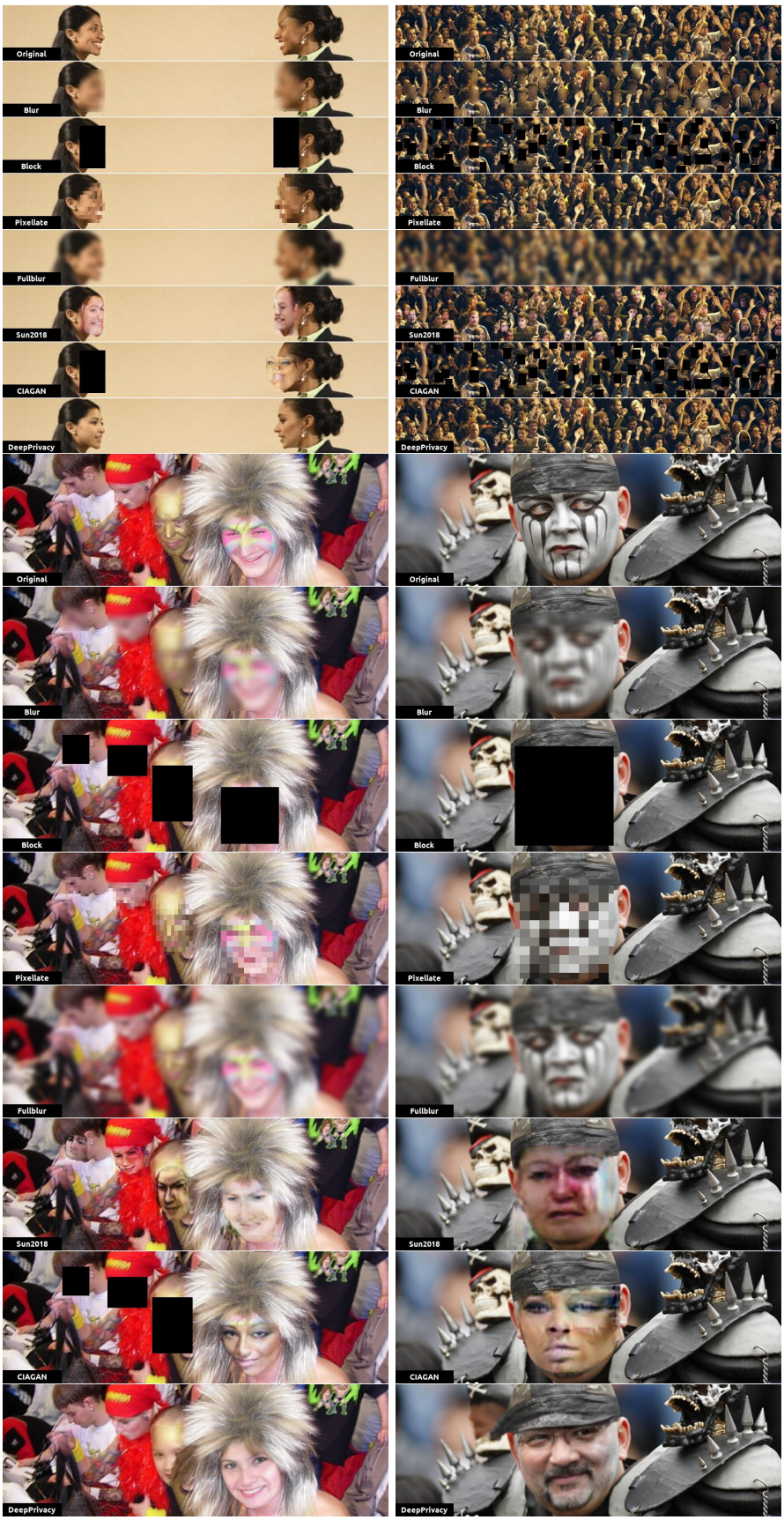

Fig. 10: Additional examples of all the employed anonymization methods, showing uncommon poses (side views), very small faces and uncommon faces such as those with face paint. Best viewed digitally and zoomed in.