DartBlur: Privacy Preservation with Detection Artifact Suppression Appendix

We give more implementation details for the proposed De-artifact Blurring (DartBlur) and report more experimental results to verify detection artifact suppression. We include the data pre-processing and training code in the supplementary material with detailed code instructions. Furthermore, We give the links to the public data sets for evaluation purposes.

A. Notations Summarization

The notations that appeared in the paper are summarized below.

Symbol	**Description**
$\boldsymbol{x}$	The original image image with RGB channels
$\boldsymbol{b}$	The binary mask obtained from ground truth bounding boxes
f	Fixed face detector pretrained with original images
g	The function of DartBlur
$\widetilde{g}$	To-be-learned U-Net-style neural network in g
f_g	On-the-fly face detector trained with blurred images
$\mathcal{G}$	Gaussian blur function
$\boldsymbol{\theta}_g$	The parameters of model g
$\boldsymbol{\theta}_{f_g}$	The parameters of model f_g
$\mathcal{L}_{\text{det}}(\cdot,\cdot)$	The loss function for face detection
$\mathcal{L}_{\text{rev}}$	The objective for review convenience
ϵ_{rev}	The threshold hyper-parameter of $\mathcal{L}_{\text{rev}}$
$\mathcal{L}_{\text{oper}}$	The objective for operation fidelity
$\mathcal{L}_{\text{post}}$	The objective for post-hoc fidelity
$\mathcal{L}_{\text{cycl}}$	The objective for cycle fidelity

B. Implementation Details

This section gives more implementation details for our DartBlur, face detectors, datasets, and training and evaluation process.

B.1. The blurring function of DartBlur

The overall structure of the trainable neural network $\widetilde{g}$ of DartBlur follows a typical U-Net style, where the first half is the feature extractor with downsampling, and the second half is the generator with upsampling. Residual connections and dense connections are also applied. The detailed structure of the network can be viewed in the code we provide.

B.2. Evaluation process

B.2.1 Dateset pre-processing

For ground-truth face bounding boxes, we filter out the boxes with negative length and width and take the rectangular intersection with the image for boxes beyond the image boundaries. We reduce the influence of pseudo-real ground truth on the detector training process by filtering the noise of the face bounding boxes.

For the FDDB dataset, we convert the ground-truth ellipse region to its horizontal outer-rectangular box as the ground truth and use it for training and evaluation. Since FDDB did not provide an official dataset split, we randomly selected 10% images for validation and others for training.

We analyzed the statistics of image sizes and the number of faces of the three datasets. The results are reported in Table 1.

B.2.2 Face detector architecture details

To evaluate the detection artifacts produced by various blurring methods, we retrain face detectors on clean and blurred datasets. The details of the three detector architectures are as follows.

- **ReinaFace**[1]**.** We chose MobileNet0.25 as backbone network which was trained on the ImageNet dataset. The codes was privoded by open source project.
- **PyramidBox**[2]. We used VGG16 as the backbone of PyramidBox. As reported by the code providers, we found that the training loss went to *NaN* when the initial learning rate was set to $1\text{e}{-3}$. So we used an SGD optimizer with a learning rate starting at $5\text{e}{-4}$. Additionally, we omitted the head detection loss.
- **YOLOv5**[3]. For YOLOv5, we chose the medium size YOLOv5m as the pretrained model, as it provided a balanced performance w.r.t memory consumption and detection performance for images around 640-pixel scale.

[1] https://github.com/biubug6/Pytorch_Retinaface

[2] https://github.com/yxlijun/Pyramidbox.pytorch

[3] https://github.com/ultralytics/yolov5

Dataset	# images	# faces	Avg. # faces per image	Avg. resolution of images	Avg. resolution of faces	# images for training	# faces for training	# images for testing	# faces for testing
WIDER FACE	32,203	393,703	12.23	1024 × 887	37 × 29	12,880	159,420	3,226	39,708
FDDB	2,845	5,171	1.82	377 × 399	140 × 94	2,560	4,641	285	530
CrowdHuman	19,370	566,453	29.24	1361 × 967	49 × 43	15,000	438,745	4,370	127,708

Table 1. The statistics of the WIDER FACE, FDDB, and CrowdHuman dataset.

Method	Strongly Agree	Weakly Agree	Weakly Disagree	Strongly Disagree
Gauss. blur	57.74%	39.95%	2.31%	0.00%
DartBlur	76.32%	22.16%	1.52%	0.00%

Table 2. Results of human evaluation of privacy protection effects for Gaussian Blur and DartBlur.

C. Additional Experimental Results

C.1. Human evaluation for privacy protection

We conducted a human evaluation to assess the effectiveness of privacy protection. We displayed 20 randomly-chosen original images and their corresponding Gaussian blur and DartBlur images to 95 individuals, and asked them to judge whether facial characteristic information is completely protected. An example of 20 questions is shown in Figure 2. Table 2 summarizes the results and shows that DartBlur has better privacy protection effectiveness.

C.2. Privacy recovering experiment

We conducted experiments to test whether private information can be recovered from DartBlur. First, the state-of-the-art GFP-GAN [4] was used to implement face restoration on DartBlur images. Then, we also trained an in-domain U-Net model to reconstruct the original (clean) images from DartBlur images on WIDER FACE. As illustrated in Eq. (1), the reconstruction function $\bar{g}$ acts only on the faces region, then we have

$$\bar{g}(g(\boldsymbol{x}, \boldsymbol{b}), \boldsymbol{b}) = \boldsymbol{x} \odot (1 - \boldsymbol{b}) + \bar{g}(g(\boldsymbol{x}, \boldsymbol{b})) \odot \boldsymbol{b}, \quad (1)$$

where $\bar{g}$ is a U-net network with the same structure as $\widetilde{g}$. We used the following loss function to try to recover the original image from DartBlur images

$$\mathcal{L} = \|\bar{g}(g(\boldsymbol{x}, \boldsymbol{b}), \boldsymbol{b}) - \boldsymbol{x}\|_1 . \quad (2)$$

Figure 1 shows four cases from the testing set of WIDER FACE. The results demonstrate that the erased privacy information cannot be trivially restored.

C.3. Detection results

We visually compare the detection artifacts of Gaussian Blur and DartBlur in Figures 3 and 4, respectively. With the "clean" RetinaFace detector f, we show the detection results on the original image at the top left and the results on the blurred image at the top right. Then, we train the "blurred" RetinaFace detector f_g, and show the detection results on the original image at the bottom left and the results on the blurred image at the bottom right. Compared to the results of Gaussian Blur in Figure 3, the results of DartBlur in Figure 4 shows significantly fewer artifacts, since the results are more aligned with the top-left part in Figure 4.

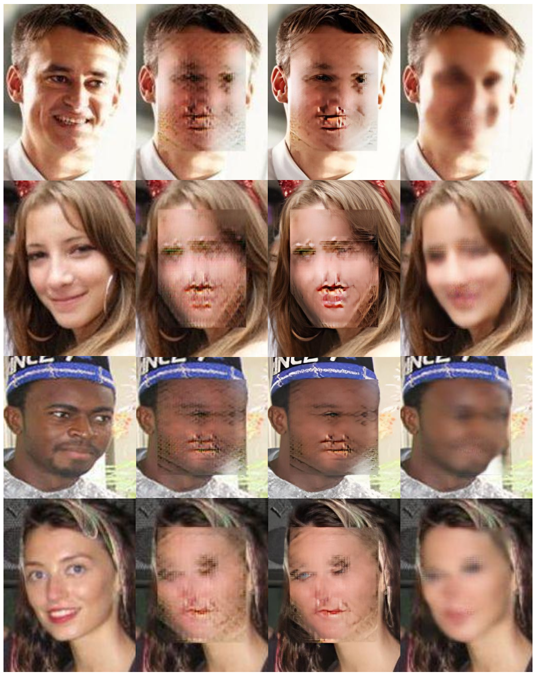
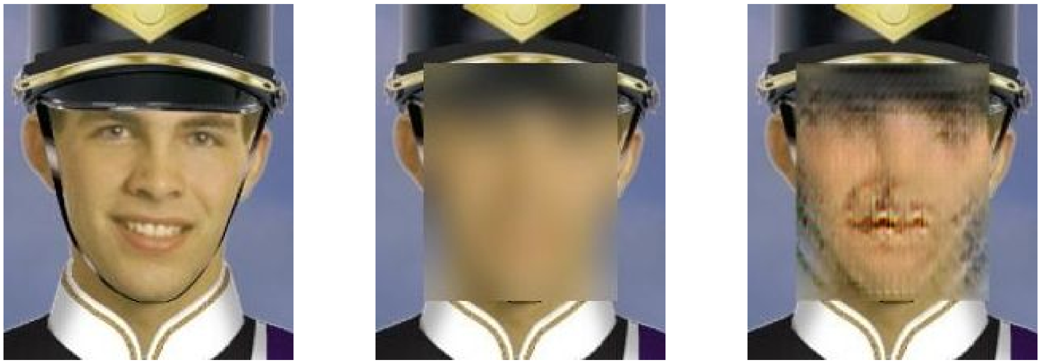


Figure 1. From left to right: the original images from WIDER FACE, the DartBlur images, the images processed with GFP-GAN on DartBlur versions, and the images processed with trained U-Net $\bar{g}$. on DartBlur versions. The results demonstrate that the erased privacy information cannot be trivially restored.

We further apply "clean" RetinaFace on different blurred images to detect faces. Figure 5 shows detection results

[4] https://github.com/TencentARC/GFPGAN



(a) Fig 1

(b) Fig 2

(c) Fig 3

Please compare the degree of privacy protection in Figs 2 and 3, based on Fig 1 (only the face areas are considered, please ignore clothing and hair), and answer the following questions.

Q 1. *Given the blurred face in Fig 2, to what degree do you agree that private information is protected?* [ ]

(1) Strongly Agree; (2) Weakly Agree; (3) Weakly Disagree; (4) Strongly Disagree.

Q 2. *Given the blurred face in Fig 3, to what degree do you agree that private information is protected?* [ ]

(1) Strongly Agree; (2) Weakly Agree; (3) Weakly Disagree; (4) Strongly Disagree.

Figure 2. An example of human evaluation.

on origin images, Gaussian blurred images, and DartBlur images. It can be observed that Gauss-Blurred images are difficult for detectors to work, but the DartBlur still retains some features of the face that detectors can perceive. Additionally, the predicted face landmarks on DartBlur images is also closer to the results on original images, as illustrated in Figure 5, even though we did not involve the landmark loss in the training process of DartBlur.

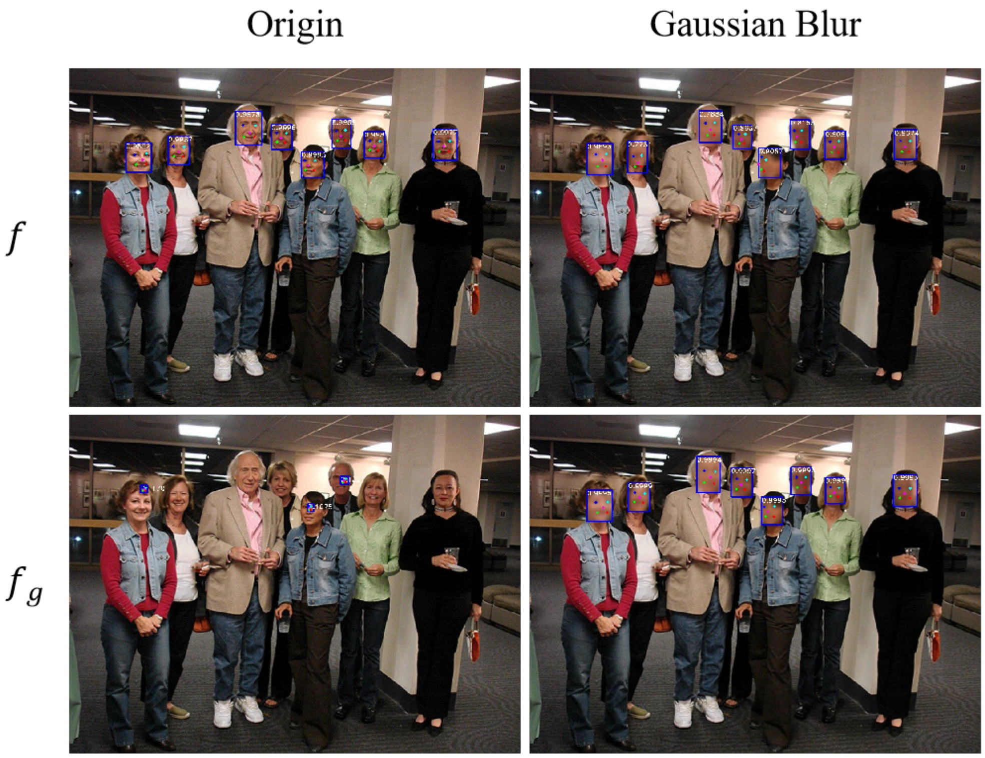


Figure 3. Detection artifacts of Gaussian Blur.

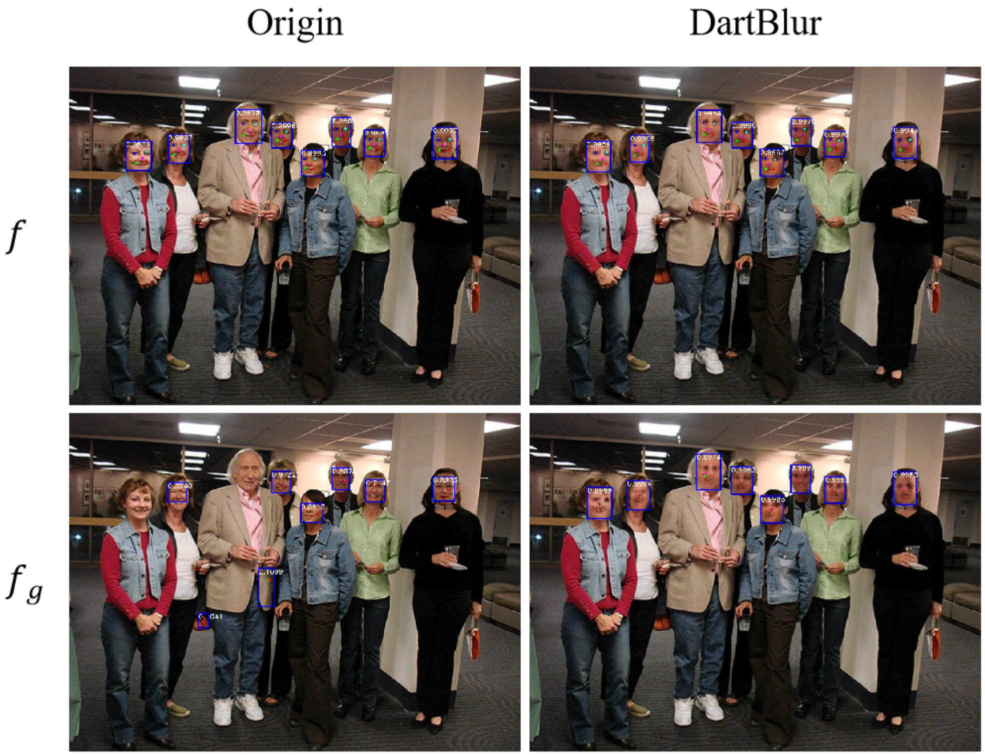


Figure 4. Detection artifacts of DartBlur.

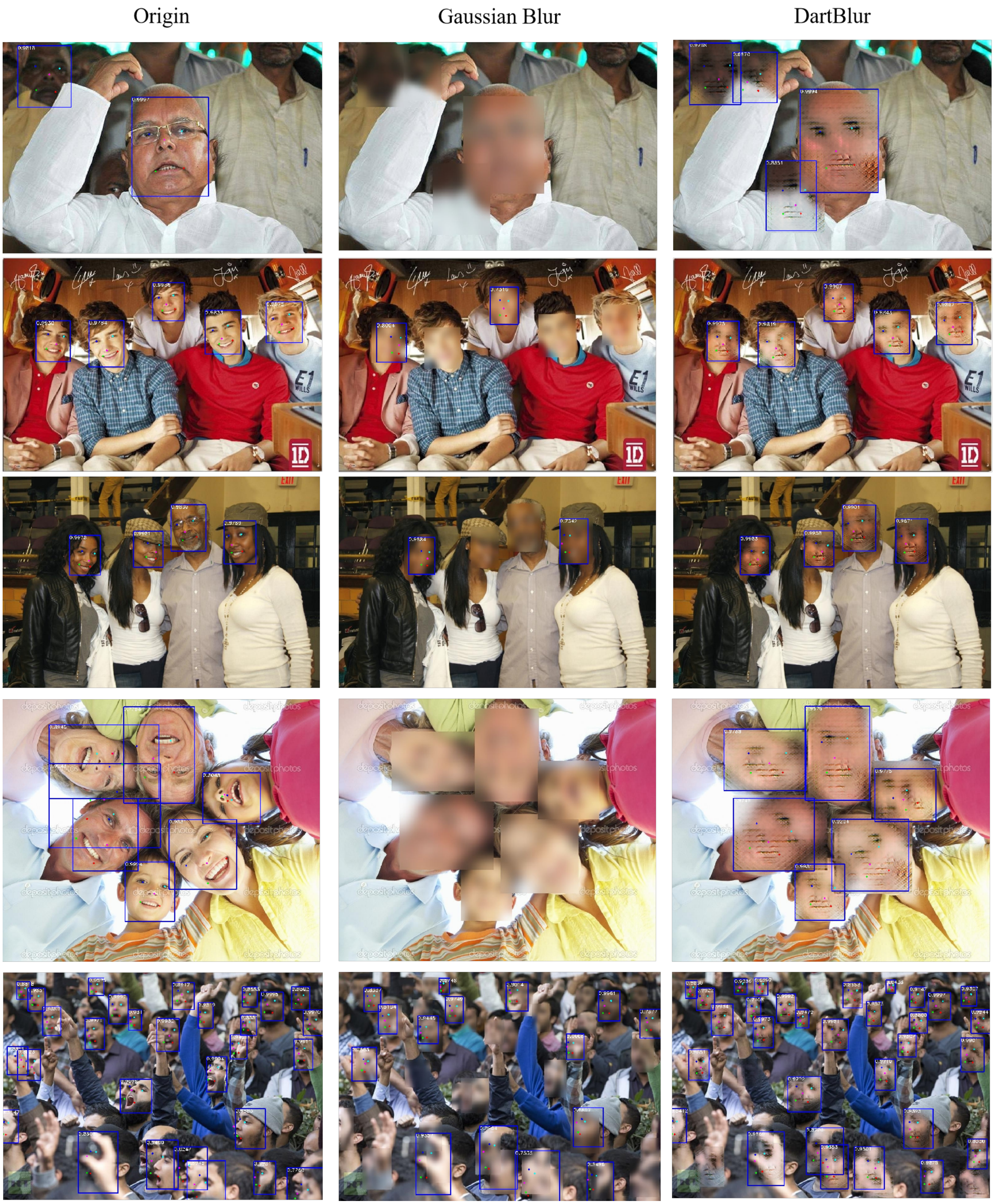


Figure 5. Comparison among detection results of “clean” RetinaFace on the WIDER FACE testing dataset. Left column: face bounding boxes detected on clean images. Middle column: detection results on Gaussian blurred images. Right column: detection results by on DartBlur images.