# DartBlur: Privacy Preservation with Detection Artifact Suppression

Baowei Jiang*, Bing Bai*, Haozhe Lin*, Yu Wang, Yuchen Guo, Lu Fang†
Tsinghua University
fanglu@tsinghua.edu.cn

## Abstract

*Nowadays, privacy issue has become a top priority when training AI algorithms. Machine learning algorithms are expected to benefit our daily life, while personal information must also be carefully protected from exposure. Facial information is particularly sensitive in this regard. Multiple datasets containing facial information have been taken offline, and the community is actively seeking solutions to remedy the privacy issues. Existing methods for privacy preservation can be divided into blur-based and face replacement-based methods. Owing to the advantages of review convenience and good accessibility, blur-based based methods have become a dominant choice in practice. However, blur-based methods would inevitably introduce training artifacts harmful to the performance of downstream tasks. In this paper, we propose a novel De-artifact Blurring (DartBlur) privacy-preserving method, which capitalizes on a DNN architecture to generate blurred faces. DartBlur can effectively hide facial privacy information while detection artifacts are simultaneously suppressed. We have designed four training objectives that particularly aim to improve review convenience and maximize detection artifact suppression. We associate the algorithm with an adversarial training strategy with a second-order optimization pipeline. Experimental results demonstrate that DartBlur outperforms the existing face-replacement method from both perspectives of review convenience and accessibility, and also shows an exclusive advantage in suppressing the training artifact compared to traditional blur-based methods. Our implementation is available at https://github.com/JaNg2333/DartBlur.*

## 1. Introduction

Computer vision (CV) technology has been influencing our daily life in many ways. However, successful CV models often have to rely on large-scale datasets collected from real-world scenes, which raises concerning privacy issues.

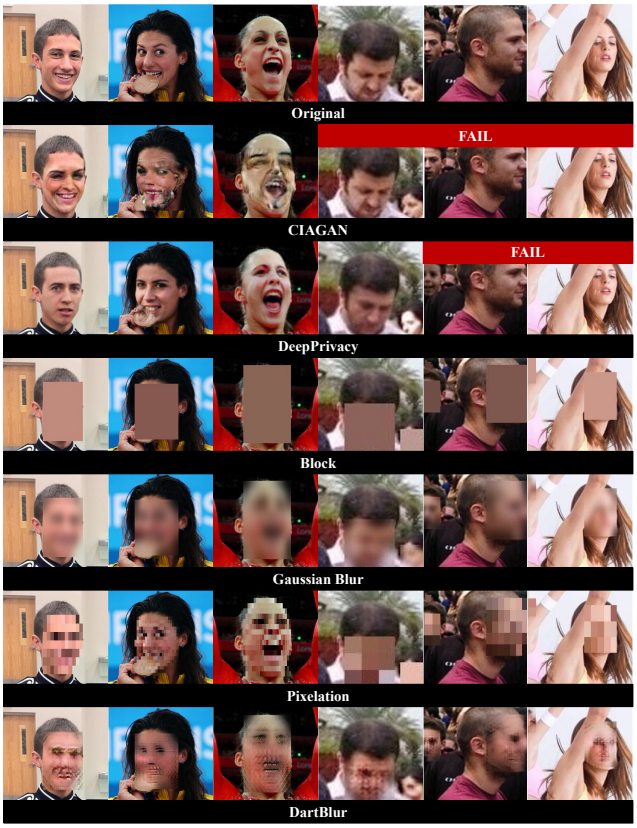


Figure 1. Example faces and anonymized versions by existing methods and DartBlur. As presented, blur-based methods facilitate review convenience, and face replacement-based methods may fail when the keypoint detector does not work as expected. Best viewed in color.

The CV community has started to take privacy issues seriously. Existing privacy-preserving methods can be divided into blur-based methods (e.g., Block, Gaussian blur, Pixelation) and face replacement-based methods (e.g., CIAGAN [24], DeepPrivacy [13], DeIdGAN [18]). Blur-based methods are simple to implement but inevitably introduce additional noise and artifacts into the actual CV task [37]. For example, Gaussian blur patterns are easier to recognize. Therefore, face detectors trained on Gaussian blurred

*These authors contributed equally to this work.
†Lu Fang is the corresponding author (www.luvision.net).

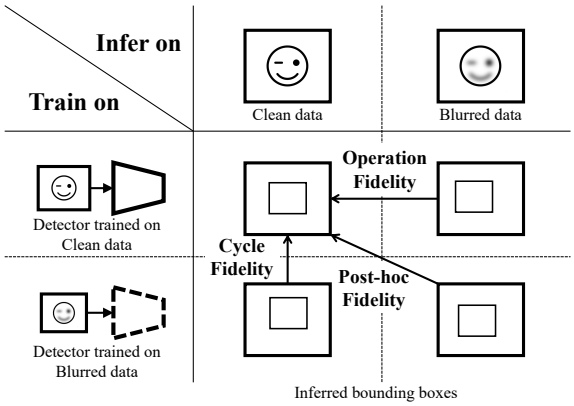


Figure 2. Illustration of detection artifact suppression. We encourage the blur model to maintain all the operation fidelity, post-hoc fidelity, and cycle fidelity.

datasets will take the shortcut to identify the blur patterns instead of the actual faces. In contrast, face replacement-based methods attempt to generate synthesized faces in order to replace the original faces through generative models such as generative adversarial networks (GANs) [7]. Such replacements tend to preserve the critical human face features that can effectively trigger the face detector to work, whereas discriminative individual identification characteristics are mostly erased.

Despite the advantages of face replacement-based methods, blur-based methods are usually still preferred in practice [1, 6, 9, 11, 26, 35, 37, 39]. In fact, blur-based methods usually make it easier to determine whether human identification is removed, while face replacement-based methods require careful face-face comparisons during an ethical review. Face replacement-based methods also hinge on the quality of landmark detection [13, 24] or semantic segmentation [18] techniques, and the generative training itself, which often requires additional face data, would also raise potential privacy issues.

The above concerns motivate us to rethink and design a novel blur-based privacy protection paradigm with the following goals: (1) **Accessibility**. The method should work well without relying on the quality of other pretrained models, such as landmark detection. (2) **Review convenience**. One can quickly determine whether or not identifiable human information is successfully concealed during an ethical review. (3) **Detection artifact suppression**. The blur function should avoid introducing much training artifacts to the detector, and specifically, we desire the following properties for detection artifact suppression, as illustrated in Figure 2.

- *Operation Fidelity*. Open-source models trained on clean data should produce similar results between clean and blurred data for model utility flexibility.
- *Post-hoc Fidelity*. The recognition performance should be maximally invariant to the images' features before and after blurring. In other words, the distance between hard cases (in terms of recognition) and easy cases on clean data should be maintained in the feature space after blurring.
- *Cycle Fidelity*: Models trained on blurred data should produce good recognition results on clean testing data.

Given the above considerations, we propose a novel privacy-preserving model called De-artifact Blurring (DartBlur). DartBlur is a learnable U-Net model [28] that is fed with Gaussian blurred images and face bounding boxes as input, and outputs detection artifact-suppressed blurred images without relying on other pretrained models like landmark detection. We propose four training objectives, each specifically addressing the mentioned concerns above, and the implementation resorts to an adversarial training strategy with a second-order optimization pipeline. Example images anonymized by existing methods and DartBlur are presented in Figure 1.

The main contributions of this paper can be summarized as follows.

- We propose a new blur-based privacy preservation model DartBlur by taking into account the actual accessibility of the model, review convenience, and detection artifact suppression simultaneously.
- DartBlur model is associated with four novel training objectives that each directly addresses the desired properties. We also design an adversarial training strategy with a second-order optimization for model training.
- We demonstrate that DartBlur can effectively protect personal privacy while suppressing detection artifacts on various benchmarks.

## 2. Related Work

We review two fields of related work, including face anonymization and adversarial attacks on machine learning models.

**Face Anonymization** Conventional face anonymization is achieved by heuristics such as Gaussian blur and pixelation [27]. These methods are easy and robust to deploy, and human reviewers can quickly determine whether privacy is protected. However, these heuristic blur-based methods destroy the features required for face detectors to work and introduce significant artifacts into the datasets. For example, Klomp *et al*. [17] find that face detectors trained on blurred faces perform poorly on clean data. Besides, there are also explorations for deblurring images [19, 25], as the functions of these heuristic methods are simple and fixed.

Recently, anonymization methods based on face replacement have emerged, including CIAGAN [24], DeepPrivacy [13], DeIdGAN [18], IdentityDP [36], FICGAN [16],

CFA-GAN [22], etc. These GAN-based methods usually first extract keypoints or conduct semantic segmentation on clean faces, then generate new faces based on the information and context, and finally, judge whether the forged face's quality meets a classification discriminator's requirements. Nevertheless, keypoint detection models are prone to errors and may break the whole pipeline. It's cumbersome to check whether face replacement-based methods anonymize the original person's identity, especially if the generated faces are highly qualified. Furthermore, training these GAN-based methods often relies on additional real face data, which raises further concerns about privacy protection.

In this paper, we propose a novel face anonymization method that inherits the merits of both blur-based and face replacement-based methods. With the proposed DartBlur, we can simultaneously achieve review convenience and effective detection artifact suppression.

**Adversarial Attacks** Existing machine learning methods, including but not limited to deep neural networks, have been shown to be vulnerable to adversarial attack [31], and adversarial attacks are found to be transferable [34]. Several models explain the existence of adversarial samples [5,8,14,23,29]. For example, Ilyas *et al*. [14] proposes that adversarial examples can be attributed to the presence of features that are highly predictive, yet brittle and incomprehensible to humans.

From the point of view of Ilyas *et al*. [14], in this paper, we preserve privacy by finding a set of predictive, transferable, and incomprehensible features that are only related to face presence instead of personal identity.

## 3. Methodology

In this section, we introduce the detailed methodology of DartBlur. We first present the overview framework, then give the training strategy in detail. The overview methodology of DartBlur is presented in Figure 3.

### 3.1. Overview framework

In this section, we present the notations and the specific training objectives to achieve the review convenience and detection artifact suppression.

**Notations** The notations used in this paper are defined as follows.

Let $\boldsymbol{x} \in \mathbb{R}^{3\times h\times w}$ represent the $h \times w$-resolution original image with 3 channels (RGB), and let $\boldsymbol{b} \in \mathbb{B}^{3\times h\times w}$ represent the binary mask computed based on ground-truth bounding boxes, where the elements of $\boldsymbol{b}$ within a bounding box are 1, and 0 otherwise. Let $g$ represent the blur function, and $g(\boldsymbol{x}, \boldsymbol{b})$ represent the blurred image. Note that we only blur the regions within bounding boxes, so

$$g(\boldsymbol{x}, \boldsymbol{b}) = \boldsymbol{x} \odot (1 - \boldsymbol{b}) + \widetilde{g}(\mathcal{G}(\boldsymbol{x}, \boldsymbol{b})) \odot \boldsymbol{b}, \tag{1}$$

where $\widetilde{g}$ is the deep neural network to be learned during training. We adopt U-Net-style fully convolutional neural networks [28] to practically implement $\widetilde{g}$. Symbol $\odot$ computes element-wise product between two tensors. Here, $\mathcal{G}(\cdot)$ is the conventional Gaussian blur function, which is applied on each image and used to blur the faces within each bounding box to hide personal identification-related signals.

Let $f$ represent the fixed face detector pretrained on clean data, and $f_g$ represent the face detector trained with blurred images $g(\boldsymbol{x}, \boldsymbol{b})$ and the ground-truth bounding boxes. In addition, we use $\boldsymbol{\theta}$ to represent the parameters of neural network models. For example, $\boldsymbol{\theta}_g$ is the parameters of model $g$, and $\boldsymbol{\theta}_{f_g}$ is the parameters of model $f_g$.

**Objective for review convenience** Human reviewers can judge whether personal identification details have been removed by blur-based methods at a glance. In comparison, when face replacement-based methods are adopted, it is usually more demanding to check if the original person identification information has been successfully replaced. In this regard, we expect the output to be as close as Gaussian blurred image in the pixel space, in order to ease the ethical review process. This motivation is achieved with the following loss function:

$$\begin{aligned}\mathcal{L}_{\text{rev}} =& \mathcal{L}_{\text{rev}}\big(g, \boldsymbol{x}, \boldsymbol{b}, \epsilon_{\text{rev}}\big) \\ =& \max(\|g(\boldsymbol{x}, \boldsymbol{b}) - \mathcal{G}(\boldsymbol{x}, \boldsymbol{b})\|_1 - \epsilon_{\text{rev}}, 0),\end{aligned} \tag{2}$$

where $\|\cdot\|_1$ computes the $\ell_1$ norm, and $\epsilon_{\text{rev}}$ is a threshold hyper-parameter. We use $\ell_1$ norm rather than other metrics (e.g., $\ell_2$ norm) to encourage the sparsity of pixel modification [2]. Note that $\ell_1$ norm is calculated only within the bounding boxes.

When $\mathcal{G}(\boldsymbol{x}, \boldsymbol{b})$ and $g(\boldsymbol{x}, \boldsymbol{b})$ are drastically different, $\mathcal{L}_{\text{rev}}$ forces them to become similar, and when the similarity is below $\epsilon_{\text{rev}}$, $\mathcal{L}_{\text{rev}}$ does not take effects. $\epsilon_{\text{rev}}$ can be considered as a budget for DartBlur to be different from Gaussian blur. We find that $\mathcal{L}_{\text{rev}}$ is beneficial for avoiding training collapse at the early training stage and encouraging the blur-like effect.

**Objective for detection artifact suppression** Open-source pretrained face detectors usually would fail to detect testing data with blurred faces. This is because conventional heuristic blur-based methods would incur a domain gap between training and testing data and therefore inevitably introduce detection artifacts. Even if we train face detectors on the blurred images, the model will take a shortcut in detecting the Gaussian blur or the pixelation pattern during the training, rather than detecting the actual facial features. This leads to inflated benchmark scores and sub-optimal detection performance when tested on practically useful clean face images. We therefore aim to suppress such detection

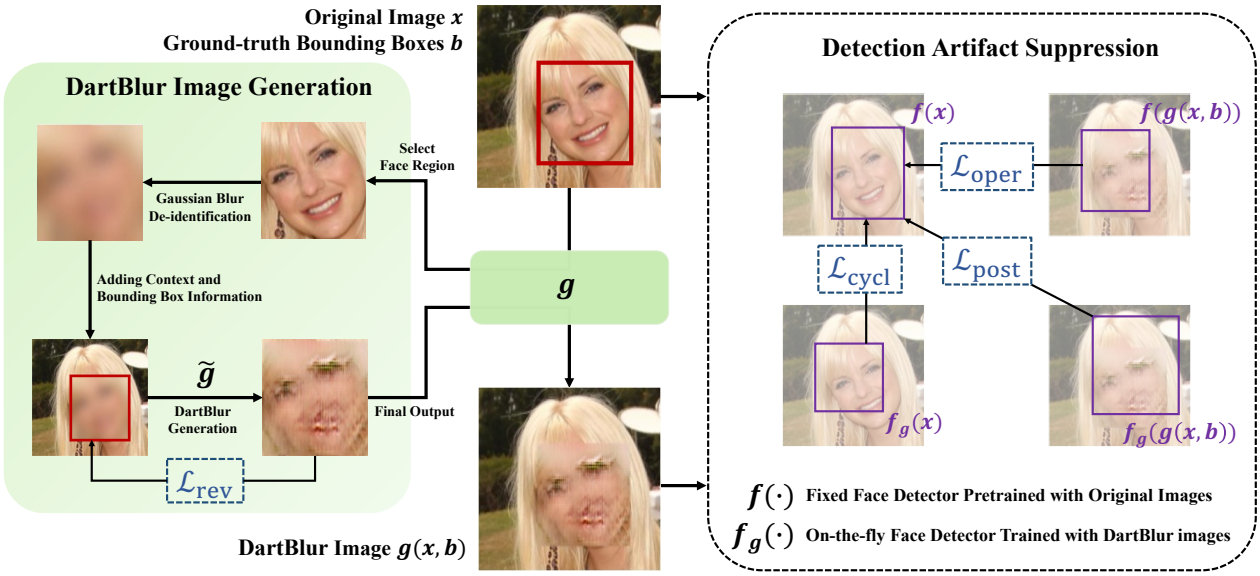


Figure 3. DartBlur takes the Gaussian blurred image and bounding boxes information as input and optimizes four objectives for review convenience and detection artifact suppression. For DartBlur image generation, we first use Gaussian blur to erase the personal identification-related signals, then use a neural network model $\widetilde{g}$ to process the regions within bounding boxes. $\mathcal{L}_{\text{rev}}$ is applied to maintain review convenience. Detection artifact suppression is achieved by $\mathcal{L}_{\text{oper}}$, $\mathcal{L}_{\text{post}}$, and $\mathcal{L}_{\text{cycl}}$. Best viewed in color.

artifacts through the following objectives:

$$\mathcal{L}_{\text{oper}} = \mathcal{L}_{\text{det}}\Big(f\big(g(\boldsymbol{x}, \boldsymbol{b})\big), f(\boldsymbol{x})\Big), \tag{3}$$

$$\mathcal{L}_{\text{post}} = \mathcal{L}_{\text{det}}\Big(f_g\big(g(\boldsymbol{x}, \boldsymbol{b})\big), f(\boldsymbol{x})\Big), \tag{4}$$

$$\mathcal{L}_{\text{cycl}} = \mathcal{L}_{\text{det}}\Big(f_g\big(\boldsymbol{x}\big), f(\boldsymbol{x})\Big), \tag{5}$$

where $\mathcal{L}_{\text{det}}(\cdot, \cdot)$ is the loss function for face detection. In this paper, we use the MultiBoxLoss [21] to compute $\mathcal{L}_{\text{det}}(\cdot, \cdot)$, and consider both the localization and classification loss[1].

Here, $\mathcal{L}_{\text{oper}}$ is the loss for operation fidelity which encourage open-source pretrained face detectors to produce similar results on both the clean and the blurred images. $\mathcal{L}_{\text{post}}$ is the loss for post-hoc fidelity which encourage the difficulty of faces to be unchanged before and after blurring processing. $\mathcal{L}_{\text{cycl}}$ is the loss for cycle fidelity which encourage the face detector trained on blurred faces capable of detecting real-world clean faces. An illustrative explanation of the fidelity metrics can be found in Figure 2. Ideally, if $f\big(g(\boldsymbol{x}, \boldsymbol{b})\big)$, $f_g\big(g(\boldsymbol{x}, \boldsymbol{b})\big)$, and $f_g\big(\boldsymbol{x}\big)$ are all exactly the same as $f\big(\boldsymbol{x}\big)$, the blur functions becomes "insenible" to detectors. So by forcing them to be close to $f\big(\boldsymbol{x}\big)$, we can suppress the detection artifacts.

**Overall training objective** We aim to optimize the blur function $g$ so that we can achieve both review convenience and detection artifact suppression simultaneously. To reach this goal, we formalize the overall training objective as the optimization problem:

$$\begin{aligned} \boldsymbol{\theta}_g^* &= \arg\min_{\boldsymbol{\theta}_g} \mathcal{L}_{\text{overall}} \\ &= \arg\min_{\boldsymbol{\theta}_g} (\mathcal{L}_{\text{rev}} + \mathcal{L}_{\text{oper}} + \mathcal{L}_{\text{post}} + \mathcal{L}_{\text{cycl}}), \end{aligned} \tag{6}$$

where $\boldsymbol{\theta}_g$ parameterizes the blurring function $g$ and is to be learnt during training. Note that specific weighting hyper-parameters can be further associated with each of the objective terms above. We nevertheless find equally weighted objectives (i.e., weighted by constant 1 as in Equation (6)) empirically provides satisfactory results. We also empirically ablate the effect of each term in Section 4.3.

### 3.2. Adversarial training strategy with second-order optimization

As discussed in Section 3.1, we aim to find the optimal $\boldsymbol{\theta}_g^*$ for $\mathcal{L}_{\text{overall}}$. However, optimizing these objectives is not straightforward. Especially, $g$ does not appear explicitly in $\mathcal{L}_{\text{cycl}}$ in Equation (5).

To overcome the difficulties, we develop an adversarial training strategy with a second-order optimization pipeline through unrolled first-order optimization loops [10]. The intuition is that model $f_g$ is trained with the blurred image $g(\boldsymbol{x}, \boldsymbol{b})$ and the ground-truth bounding box $\boldsymbol{b}$, so we can track the high-order influence from $g$ to $f_g$ to $\mathcal{L}_{\text{cycl}}$, and thus backward gradient from $\mathcal{L}_{\text{cycl}}$ to $g$.

Formally, when training the detector $f_g$ based on given input $g(\boldsymbol{x}, \boldsymbol{b})$ and target $\boldsymbol{b}$, the model's parameters $\boldsymbol{\theta}_{f_g}$ be-

[1]Since we use $f(x)$ as the target, the loss for landmark detection is abandoned.

**Algorithm 1: Training Algorithm for DartBlur**

**Input:** Dataset $\mathcal{D}$, pretrained face detector $f$
**Hyper-parameter:** Step size $\alpha$ and $\beta$, threshold $\epsilon_{\text{rev}}$
**Output:** Optimized parameters $\boldsymbol{\theta}_g^*$

1 Randomly initialize $\boldsymbol{\theta}_g$ and $\boldsymbol{\theta}_{f_g}$;
2 **while** *not converge* **do**
3 Sample a batch of data $\boldsymbol{x}, \boldsymbol{b} \sim \mathcal{D}$;
`// Optimize g with f_g fixed`
4 Update $\boldsymbol{\theta}_g \leftarrow \boldsymbol{\theta}_g - \beta \nabla_{\boldsymbol{\theta}_g} \mathcal{L}_{\text{rev}}(g, \boldsymbol{x}, \boldsymbol{b}, \epsilon_{\text{rev}})$;
5 Update $\boldsymbol{\theta}_g \leftarrow \boldsymbol{\theta}_g - \beta \nabla_{\boldsymbol{\theta}_g} \mathcal{L}_{\text{det}}\Big(f\big(g(\boldsymbol{x}, \boldsymbol{b})\big), f(\boldsymbol{x})\Big)$;
6 Update $\boldsymbol{\theta}_g \leftarrow \boldsymbol{\theta}_g - \beta \nabla_{\boldsymbol{\theta}_g} \mathcal{L}_{\text{det}}\Big(f_g\big(g(\boldsymbol{x}, \boldsymbol{b})\big), f(\boldsymbol{x})\Big)$;
`// Optimize g considering second-order effects`
7 Compute adapted parameters with gradient descent:
$\boldsymbol{\theta}'_{f_g} = \boldsymbol{\theta}_{f_g} - \alpha \nabla_{\boldsymbol{\theta}_{f_g}} \mathcal{L}_{\text{det}}\Big(f_g\big(g(\boldsymbol{x}, \boldsymbol{b})\big), \boldsymbol{b}\Big)$;
8 Update $\boldsymbol{\theta}_g \leftarrow \boldsymbol{\theta}_g - \beta \nabla_{\boldsymbol{\theta}_g} \mathcal{L}_{\text{det}}\big(f_g(\boldsymbol{x}; \boldsymbol{\theta}'_{f_g}), f(\boldsymbol{x})\big)$;
`// Optimize f_g with g fixed`
9 Update $\boldsymbol{\theta}_{f_g} \leftarrow \boldsymbol{\theta}_{f_g} - \beta \nabla_{\boldsymbol{\theta}_{f_g}} \mathcal{L}_{\text{det}}\Big(f_g\big(g(\boldsymbol{x}, \boldsymbol{b})\big), \boldsymbol{b}\Big)$;
10 **end**

come $\boldsymbol{\theta}'_{f_g}$. For example, when using one gradient update,

$$\boldsymbol{\theta}'_{f_g} = \boldsymbol{\theta}_{f_g} - \alpha \nabla_{\boldsymbol{\theta}_{f_g}} \mathcal{L}_{\text{det}}\Big(f_g\big(g(\boldsymbol{x}, \boldsymbol{b}); \boldsymbol{\theta}_{f_g}\big), \boldsymbol{b}\Big), \tag{7}$$

where $\alpha$ is the step size.

Equation (7) links the adapted parameters of $f_g$, i.e., $\boldsymbol{\theta}'_{f_g}$ and model $g$. Then, we can derive the gradient of $\mathcal{L}_{\text{cycl}}$ w.r.t $\boldsymbol{\theta}_g$, i.e.,

$$\begin{aligned}
&\nabla_{\boldsymbol{\theta}_g} \mathcal{L}_{\text{cycl}} \\
=&\nabla_{\boldsymbol{\theta}_g} \mathcal{L}_{\text{det}}\Big(f_g\big(\boldsymbol{x}; \boldsymbol{\theta}'_{f_g}\big), f(\boldsymbol{x})\Big) \\
=&\nabla_{\boldsymbol{\theta}_g} \mathcal{L}_{\text{det}}\Big(f_g\big(\boldsymbol{x}; \boldsymbol{\theta}_{f_g} - \alpha \nabla_{\boldsymbol{\theta}_{f_g}} \mathcal{L}_{\text{det}}\Big(f_g\big(g(\boldsymbol{x}, \boldsymbol{b}); \boldsymbol{\theta}_{f_g}\big), \boldsymbol{b}\Big)\big), f(\boldsymbol{x})\Big).
\end{aligned} \tag{8}$$

Equipped with the above analyses, we propose the adversarial training strategy with a second-order optimization for DartBlur. The full algorithm, in the general case, is outlined in Algorithm 1. Note that model $f$ is pretrained and keeps fixed, while model $f_g$ is trained on the fly. When training $f_g$, we use the ground-truth bounding boxes $\boldsymbol{b}$ as the target, while when training model $g$, we use the output of the pretrained detector, i.e., $f(\boldsymbol{x})$. Thus the training procedure follows an adversarial style.

### 3.3. Model training details

In practice, we first run only Line 4, Line 5, and Line 9 for warming up and then use the checkpoints to initialize model $g$ and $f_g$. This trick is beneficial for speeding up convergence. Besides, we insert two epochs optimizing $f_g$ by running only Line 9 between epochs optimizing both model $g$ and $f_g$. This is essential for the backward of up-to-date gradient from $f_g$, as, during our experiments, $f_g$ cannot catch up with $g$ only with on-the-fly training. Furthermore, we employ label smoothing on the classification loss when training $f_g$ to prevent overconfidence and avoid gradient vanishing.

For the weights of objectives in eq:overall, i.e., $\mathcal{L}_{\text{rev}}$, $\mathcal{L}_{\text{oper}}$, $\mathcal{L}_{\text{post}}$, and $\mathcal{L}_{\text{cycl}}$, we find that equal weights worked fine. We did not deliberately adjust the weights. Advanced multi-task learning methods like GradNorm [3] and Pareto multi-task learning [20] may further improve the results. We leave it as future work.

The model architecture information and other details are put in the appendix.

## 4. Experiments

In this section, we report the experimental results in details. We first introduce the experimental settings, then show that DartBlur achieved privacy protection and state-of-the-art fidelity among blur-based methods. Moreover, DartBlur also reserved the property of review convenience, and the learned blur function was transferable between different datasets and face detector architectures. We also present the ablation study results to demonstrate the components' effectiveness.

### 4.1. Experimental settings

We first introduce the experimental settings, including the used dataset, the baseline methods, the evaluation protocol and metrics, and other implementation details.

**Dataset** We performed experiments on the following public datasets:

- **WIDER FACE [38]**. The dataset provides 16k images and 199k faces with bounding boxes. As the annotations for the testing set are not released, we used the given validation set for testing. WIDER FACE has a wide range of variation in scale, pose, illumination, expression, and occlusion.
- **FDDB [15]**. The dataset contains 2845 images with 5171 faces in different poses, resolutions, rotations, and shading.
- **Crowd Human [30]**. The dataset contains 15k images for the training set and 4,370 for the validation set, and each image includes 23 people on average. Due to the same reason with WIDER FACE, we used the validation set for testing.

**Baselines** We considered the following blur-based baselines for quantitative comparisons.

- **Block**. We averaged the pixels of each channel within the given bounding boxes.

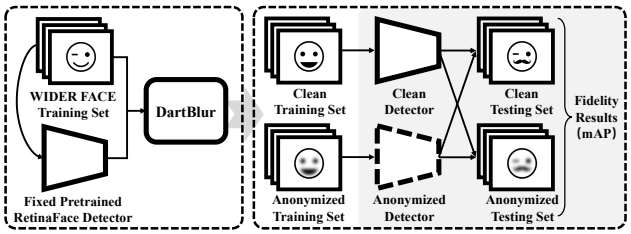


Figure 4. Illustration of our evaluation protocol. DartBlur was trained only with the training set of WIDER FACE and the RetinaFace detector while evaluated with multiple face detection datasets and model architectures.

- **Blur (Gaussian blur)**. We used a flexible Gaussian kernel to blur the face, and the shape of the Gaussian kernel was $1/2$ of the face bounding box. This Gaussian blur function was also used in DartBlur, as described in Section 3.1.
- **Pixel. (Pixelation)**. Similar to Gaussian blur, we resized the image to $1/16$ of the original shape and then resized it back to the original image size by nearest neighbor interpolation.

We did not include face replacement-based methods due to the following reasons. We found that CIAGAN and DeepPrivacy failed to replace many faces in the datasets. For example, DeepPrivacy failed to anonymize 36.7% of faces in the training set of WIDER FACE. Besides, face replacement-based methods require careful face-to-face comparison for a privacy protection review, while blur-based methods can be more review-friendly. We made an empirical study on the face replacement-based methods in Section 4.5.

**Evaluation protocol and metrics** For DartBlur training, we first trained a face detector (i.e., model $f$) with the clean training set of WIDER FACE, then we obtained the trained DartBlur following Algorithm 1.

For evaluation, we used the following protocol to simulate how DartBlur would be applied in the real world. Given a dataset for face detection, we trained a new detector on the clean training set, anonymized the dataset with DartBlur, and then trained an anonymized detector on the blurred training set. Then we evaluated the detection artifact suppression on the clean and anonymized testing set. Specifically, we used the predicted bounding boxes by the clean detector on the clean testing set as proxy ground truth and used the commonly-used metric mean average precision (mAP) to evaluate the predictions corresponding to operation fidelity, post-hoc fidelity, and cycle fidelity respectively. We use the official split of WIDER FACE and Crowd Human. For FDDB which did not provide an official training-testing split, we randomly selected 10% images for testing. Figure 4 illustrates our evaluation protocol.

To comprehensively evaluate the performance of DartBlur, we used multiple datasets for cross-dataset generalization evaluation and multiple structures of face detectors for cross-architecture generalization evaluation.

| Dataset | Fidelity | Block | Blur | Pixel. | DartBlur |
|---|---|---|---|---|---|
| WIDER FACE | Oper. Fid. | 19.20 | 83.10 | 46.35 | **96.76** |
| | Post-hoc Fid. | 77.77 | 84.04 | 80.08 | **84.70** |
| | Cycle Fid. | 0.10 | 1.97 | 7.64 | **47.22** |
| FDDB | Oper. Fid. | 4.63 | 89.00 | 4.82 | **98.06** |
| | Post-hoc Fid. | 85.49 | 93.41 | 88.34 | **94.64** |
| | Cycle Fid. | 0.00 | 0.05 | 0.02 | **41.93** |
| CrowdHuman | Oper. Fid. | 16.69 | 75.74 | 45.26 | **81.64** |
| | Post-hoc Fid. | 58.23 | 57.51 | 59.60 | **62.52** |
| | Cycle Fid. | 0.83 | 36.72 | 28.75 | **45.11** |

Table 1. Evaluation results of detection artifact suppression and cross-dataset transferability. "%" is omitted. DartBlur trained with WIDER FACE successfully suppressed detection artifacts, and was generalizable across different datasets.

**Implementation details** For the training stage, we used RetinaFace [4] with backbone MobileNet0.25 [12] as the detector (model $f$). The model was trained with the clean WIDER FACE training set by an SGD optimizer with a learning rate starting at $1e{-}2$, and cosine decay. When training DartBlur, as discussed in Section 3.1, we used the output bounding boxes of model $f$ on the clean dataset as the adversarial target, where the IoU threshold was set to 0.45, and the confidence threshold was set to 0.5. The weights for the objectives in Equation (6) were set to 1, and hyperparameter $\epsilon_{rev}$ in Equation (2) was set to 20. We used AdaM optimizer for training DartBlur. The learning rate started at $2e{-}4$ and decayed at 0.925 every epoch. For the training of on-the-fly detector $f_g$, the learning rate started at $5e{-}4$ and decayed at 0.925 every epoch, and we used a label smoothing of 0.2 on the classification loss of $f_g$. We trained DartBlur for 60 epochs in total.

For the evaluation stage, apart from RetinaFace, we considered PyramidBox [32] and YOLOv5[2] to test the cross-architecture generalization performance. All the detectors and hyperparameters were used out of the box. When training PyramidBox, we used an SGD optimizer with momentum 0.9 and a learning rate starting at $5e{-}4$. For YOLOv5, we used an SGD optimizer with learning rate starting at $1e{-}2$ and cosine decay. For training RetinaFace, we used the same hyperparameters as in the training stage.

During our experiments, all the images were resized to $768 \times 768$ for processing.

## 4.2. Performance evaluation

In this section, we report the quantitative experimental results w.r.t DartBlur. We first show that DartBlur successfully protected privacy and suppressed the detection artifacts. Then we report the results of cross-dataset and cross-

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


Figure 5. Example faces anonymized with DartBlur. (A) Examples of front faces. (B) Examples of side faces. (C) Examples of minority cases. Best viewed in color.

**(A) Original Image** **(B) Gaussian Blur Image** **(C) DartBlur Image**

Figure 6. Example image anonymized with Gaussian blur and DartBlur. On top of Gaussian blur, DartBlur tends to adjust the contrast of the image and add special textures in key areas. Besides, DartBlur only uses Gaussian blur as the preprocessing tool so that accessibility can be guaranteed. Best viewed in color.

| Architecture | Fidelity | Block | Blur | Pixel. | DartBlur |
|---|---|---|---|---|---|
| PyramidBox | Oper. Fid. | 21.34 | 84.60 | 30.55 | **95.18** |
| | Post-hoc Fid. | 75.04 | 70.59 | 65.18 | **75.16** |
| | Cycle Fid. | 0.01 | 1.70 | 10.98 | **24.68** |
| YOLOv5 | Oper. Fid. | 35.93 | 84.84 | 35.22 | **96.17** |
| | Post-hoc Fid. | 85.68 | 87.44 | 86.01 | **91.72** |
| | Cycle Fid. | 0.21 | 10.00 | 0.36 | **37.15** |

Table 2. Cross-architecture transferability evaluation on WIDER FACE. "%" is omitted. DartBlur beat baselines significantly w.r.t detection artifact suppression metrics.

architecture generalization to show that DartBlur trained with RetinaFace on the WIDER FACE dataset was well generalizable. We put the experiment to reconstruct original images from DartBlur in the appendix.

**Privacy protection** We first evaluated the effectiveness of privacy protection, by collecting human evaluation results from 95 individuals who checked 20 randomly selected images and their DartBlur versions. The results demonstrate that 98.48% of participants agreed with the notion that private information has been effectively protected. Please refer to more detailed information in the appendix.

**Detection artifact suppression** We also evaluated the effectiveness of detection artifact suppression, and the results are reported in the first row-block of Table 1. From the table, we can find that Block performed the worst, which is not surprising as Block wiped out nearly all the information within bounding boxes, thus bringing significant artifacts into the dataset. Detectors trained with Block will detect regions without any texture, thus completely failing at detecting real-world faces. Blur and Pixel. maintain some of the necessary textual. Therefore face detector trained on a clean dataset can work on the data to some extent. However, when training new detectors with the processed data, detectors still tend to capture the artifacts and perform poorly on clean data. On the other hand, DartBlur obtained the best results on all three fidelity metrics, indicating that detection artifacts have been suppressed.

**Cross-dataset transferability** To test whether the DartBlur model trained on WIDER FACE could generalize to other datasets, we involved the FDDB and CrowdHuman datasets. Experimental results are reported in the last two row blocks of Table 1. The conclusion remained the same with WIDER FACE, i.e., DartBlur obtained the best results among all baseline methods, indicating that the model could be well generalized across different datasets for detection.

**Cross-architecture transferability** During our experiments, we used RetinaFace to train DartBlur. We employed

| Ablation | Oper. Fid. | Post-hoc Fid. | Cycle Fid. |
|---|---|---|---|
| DartBlur (complete version) | 96.76 | 84.70 | 47.22 |
| w/o $\mathcal{L}_{\text{oper}}$ | 81.65 | 84.08 | 25.87 |
| w/o $\mathcal{L}_{\text{post}}$ | 97.23 | 79.34 | 8.95 |
| w/o $\mathcal{L}_{\text{cycl}}$ | 96.63 | 84.78 | 35.13 |

Table 3. Ablation study on WIDER FACE. "%" is omitted.

PyramidBox and YOLOv5 to study the cross-architecture transferability, and the results are reported in Table 2.

From the table, we find that different architectures mildly impacted the results. Compared with the results with RetinaFace on WIDER FACE in Table 1, the results slightly decreased but can still beat the heuristical blur-based methods regarding operation fidelity, post-hoc fidelity, and cycle fidelity. A straightforward extension to improve cross-architecture transferability is to use an ensemble of face detectors with different architectures during DartBlur's training [33]. We leave it as future work for exploration.

### 4.3. Ablation study

DartBlur jointly optimizes 4 loss functions during training. Apart from $\mathcal{L}_{\text{rev}}$ which encourages review convenience, we conducted the ablation study on all of the other three components and report the results in Table 3. Experimental results demonstrate the effectiveness of each objective component. Intriguingly, we observe that cycle fidelity slumps if any of the proposed objectives are absent, justifying the benefit of the jointly optimized model as a whole.

### 4.4. Case study

To intuitively demonstrate the effectiveness of DartBlur, we conducted a case study, and present the example faces anonymized with DartBlur in Figure 5, as well as the example image annonymized with both Gaussian blur and DartBlur in Figure 6. From the figures, we can find that DartBlur tends to darken the eye area, and add special textures around the eye and mouth area. The textures erase the particular shape of eyes and mouths but retain the necessary feature for detectors to work.

### 4.5. Analyses on face replacement-based methods

We conducted an empirical study on two face replacement-based methods that official implementations with trained models are publicly available, i.e., CIAGAN and DeepPrivacy. Our experiments showed that CIAGAN and DeepPrivacy could not successfully replace many faces in the WIDER FACE dataset. For example, DeepPrivacy failed to anonymize about 36.7% faces (56,898 out of 159,420) in the training set of WIDER FACE. And we found that DeepPrivacy's failure cases basically stumped CIAGAN too, since DeepPrivacy needs 7 keypoints to work, while CIAGAN requires 27 keypoints. Besides, it has been reported that the minimum face resolution required for anonymization to be applicable is around 14 pixels wide for DeepPrivacy and about 50 pixels wide for CIAGAN [17].


Figure 7. Example faces that neither CIAGAN nor DeepPrivacy could replace in. The failure cases can be divided into four categories, i.e., stylization, complex pose, distortion, and others.

We summarize some of the failure cases in Figure 7. The failure cases can be divided into four categories: stylization, complex pose, distortion, and others (mainly occluded by other objects or cropped at the edges of images). In contrast, DartBlur only uses Gaussian blur as the preprocessing tool and thus does not struggle with these cases.

## 5. Conclusion and Future Work

The privacy issue has already become a concern of the CV community. For privacy preservation, blur-based and face replacement-based methods have been developed. However, traditional heuristic blur-based and face replacement-based methods both have their shortcomings. We find that accessibility and review convenience are considered more important in practice, so blur-based methods are more widely adopted. This paper proposes DartBlur, a novel blur-based approach that balances accessibility, review convenience, and detection artifact suppression. Experiments show that DartBlur successfully achieves the design goals and has good generalization ability across datasets and architectures. In the future, we plan to explore more effective optimization methods to improve DartBlur further. We hope this work inspires researchers to develop more accessible and efficient solutions to minimize the negative impact on CV models while preserving privacy.

**Acknowledgements** This work is supported in part by Natural Science Foundation of China (NSFC) under contract No. 62125106, 61860206003, and 62088102, in part by Ministry of Science and Technology of China under contract No. 2021ZD0109901, in part by Young Elite Scientists Sponsorship Program by CAST under contract No. 2022QNRC001.

## References


[1] Holger Caesar, Varun Bankiti, Alex H Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuScenes: A multimodal dataset for autonomous driving. In *CVPR*, pages 11621–11631, 2020. 2

[2] Emmanuel J Candes, Michael B Wakin, and Stephen P Boyd. Enhancing sparsity by reweighted $\ell_1$ minimization. *Journal of Fourier analysis and applications*, 14(5):877–905, 2008. 3

[3] Zhao Chen, Vijay Badrinarayanan, Chen-Yu Lee, and Andrew Rabinovich. GradNorm: Gradient normalization for adaptive loss balancing in deep multitask networks. In *ICML*, pages 794–803. PMLR, 2018. 5

[4] Jiankang Deng, Jia Guo, Evangelos Ververas, Irene Kotsia, and Stefanos Zafeiriou. Retinaface: Single-shot multi-level face localisation in the wild. In *CVPR*, pages 5203–5212, 2020. 6

[5] Alhussein Fawzi, Hamza Fawzi, and Omar Fawzi. Adversarial vulnerability for any classifier. In *NeurIPS*, 2018. 3

[6] Andrea Frome, German Cheung, Ahmad Abdulkader, Marco Zennaro, Bo Wu, Alessandro Bissacco, Hartwig Adam, Hartmut Neven, and Luc Vincent. Large-scale privacy protection in google street view. In *ICCV*, pages 2373–2380. IEEE, 2009. 2

[7] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *Communications of the ACM*, 63(11):139–144, 2020. 2

[8] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. In *ICLR*, 2015. 3

[9] Kristen Grauman, Andrew Westbury, Eugene Byrne, Zachary Chavis, Antonino Furnari, Rohit Girdhar, Jackson Hamburger, Hao Jiang, Miao Liu, Xingyu Liu, et al. Ego4D: Around the world in 3,000 hours of egocentric video. In *CVPR*, pages 18995–19012, 2022. 2

[10] Edward Grefenstette, Brandon Amos, Denis Yarats, Phu Mon Htut, Artem Molchanov, Franziska Meier, Douwe Kiela, Kyunghyun Cho, and Soumith Chintala. Generalized inner loop meta-learning. *arXiv preprint arXiv:1910.01727*, 2019. 4

[11] Jianhua Han, Xiwen Liang, Hang Xu, Kai Chen, Lanqing Hong, Jiageng Mao, Chaoqiang Ye, Wei Zhang, Zhenguo Li, Xiaodan Liang, et al. SODA10M: A large-scale 2d self/semi-supervised object detection dataset for autonomous driving. *arXiv preprint arXiv:2106.11118*, 2021. 2

[12] Andrew G Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*, 2017. 6

[13] Håkon Hukkelås, Rudolf Mester, and Frank Lindseth. DeepPrivacy: A generative adversarial network for face anonymization. In *ISVC*, pages 565–578. Springer, 2019. 1, 2

[14] Andrew Ilyas, Shibani Santurkar, Dimitris Tsipras, Logan Engstrom, Brandon Tran, and Aleksander Madry. Adversarial examples are not bugs, they are features. *NeurIPS*, 32, 2019. 3

[15] Vidit Jain and Erik Learned-Miller. FDDB: A benchmark for face detection in unconstrained settings. Technical Report UM-CS-2010-009, University of Massachusetts, Amherst, 2010. 5

[16] Yonghyun Jeong, Jooyoung Choi, Sungwon Kim, Youngmin Ro, Tae-Hyun Oh, Doyeon Kim, Heonseok Ha, and Sungroh Yoon. Ficgan: Facial identity controllable gan for de-identification. *arXiv preprint arXiv:2110.00740*, 2021. 2

[17] Sander R Klomp, Matthew Van Rijn, Rob GJ Wijnhoven, Cees GM Snoek, and Peter HN De With. Safe fakes: Evaluating face anonymizers for face detectors. In *FG*, pages 1–8. IEEE, 2021. 2, 8

[18] Zhenzhong Kuang, Huigui Liu, Jun Yu, Aikui Tian, Lei Wang, Jianping Fan, and Noboru Babaguchi. Effective de-identification generative adversarial network for face anonymization. In *ACM MM*, pages 3182–3191, 2021. 1, 2

[19] Deepa Kundur and Dimitrios Hatzinakos. Blind image deconvolution. *IEEE Signal Processing Magazine*, 13(3):43–64, 1996. 2

[20] Xi Lin, Hui-Ling Zhen, Zhenhua Li, Qing-Fu Zhang, and Sam Kwong. Pareto multi-task learning. In *NeurIPS*, pages 12037–12047, 2019. 5

[21] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C. Berg. SSD: Single shot multibox detector. In *ECCV*, pages 21–37, 2016. 4

[22] Tianxiang Ma, Dongze Li, Wei Wang, and Jing Dong. Cfa-net: Controllable face anonymization network with identity representation manipulation. *arXiv preprint arXiv:2105.11137*, 2021. 3

[23] Saeed Mahloujifar, Dimitrios I Diochnos, and Mohammad Mahmoody. The curse of concentration in robust learning: Evasion and poisoning attacks from concentration of measure. In *AAAI*, pages 4536–4543, 2019. 3

[24] Maxim Maximov, Ismail Elezi, and Laura Leal-Taixé. CIAGAN: Conditional identity anonymization generative adversarial networks. In *CVPR*, pages 5447–5456, 2020. 1, 2

[25] Jinshan Pan, Wenqi Ren, Zhe Hu, and Ming-Hsuan Yang. Learning to deblur images with exemplars. *PAMI*, 41(6):1412–1425, 2018. 2

[26] AJ Piergiovanni and Michael Ryoo. AViD dataset: Anonymized videos from diverse countries. *NeurIPS*, 33:16711–16721, 2020. 2

[27] Slobodan Ribaric, Aladdin Ariyaeeinia, and Nikola Pavesic. De-identification for privacy protection in multimedia content: A survey. *Signal Processing: Image Communication*, 47:131–151, 2016. 2

[28] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-Net: Convolutional networks for biomedical image segmentation. In *MICCAI*, pages 234–241. Springer, 2015. 2, 3

[29] Adi Shamir, Itay Safran, Eyal Ronen, and Orr Dunkelman. A simple explanation for the existence of adversar-

ial examples with small hamming distance. *arXiv preprint arXiv:1901.10861*, 2019. 3

[30] Shuai Shao, Zijian Zhao, Boxun Li, Tete Xiao, Gang Yu, Xiangyu Zhang, and Jian Sun. Crowdhuman: A benchmark for detecting human in a crowd. *arXiv preprint arXiv:1805.00123*, 2018. 5

[31] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. In *ICLR*, 2014. 3

[32] Xu Tang, Daniel K Du, Zeqiang He, and Jingtuo Liu. PyramidBox: A context-assisted single shot face detector. In *ECCV*, pages 797–813, 2018. 6

[33] Florian Tramèr, Alexey Kurakin, Nicolas Papernot, Ian Goodfellow, Dan Boneh, and Patrick McDaniel. Ensemble adversarial training: Attacks and defenses. In *ICLR*, 2018. 8

[34] Florian Tramèr, Nicolas Papernot, Ian Goodfellow, Dan Boneh, and Patrick McDaniel. The space of transferable adversarial examples. *arXiv preprint arXiv:1704.03453*, 2017. 3

[35] Xueyang Wang, Xiya Zhang, Yinheng Zhu, Yuchen Guo, Xiaoyun Yuan, Liuyu Xiang, Zerun Wang, Guiguang Ding, David Brady, Qionghai Dai, and Lu Fang. PANDA: A gigapixel-level human-centric video dataset. In *CVPR*, pages 3268–3278, 2020. 2

[36] Yunqian Wen, Bo Liu, Ming Ding, Rong Xie, and Li Song. IdentityDP: Differential private identification protection for face images. *Neurocomputing*, 501:197–211, 2022. 2

[37] Kaiyu Yang, Jacqueline H Yau, Li Fei-Fei, Jia Deng, and Olga Russakovsky. A study of face obfuscation in ImageNet. In *ICML*, pages 25313–25330, 2022. 1, 2

[38] Shuo Yang, Ping Luo, Chen-Change Loy, and Xiaoou Tang. WIDER FACE: A face detection benchmark. In *CVPR*, pages 5525–5533, 2016. 5

[39] Shijie Yu, Shihua Li, Dapeng Chen, Rui Zhao, Junjie Yan, and Yu Qiao. COCAS: A large-scale clothes changing person dataset for re-identification. In *CVPR*, pages 3400–3409, 2020. 2