People Taking Photos That Faces Never Share: Privacy Protection and Fairness Enhancement from Camera to User

Junjie Zhu[1], Lin Gu[2], Xiaoxiao Wu[1], Zheng Li[3], Tatsuya Harada[5], Yingying Zhu[4*]

[1]Shenzhen University
[2]RIKEN, The University of Tokyo
[3]Stockton University
[4]University of Texas, Arlington
[5]The University of Tokyo, RIKEN
zhujun4jie2021@email.szu.edu.cn, yingying.zhu@uta.edu, xxwu.eesissi@szu.edu.cn

Abstract

The soaring number of personal mobile devices and public cameras poses a threat to fundamental human rights and ethical principles. For example, the stolen of private information such as face image by malicious third parties will lead to catastrophic consequences. By manipulating appearance of face in the image, most of existing protection algorithms are effective but irreversible. Here, we propose a practical and systematic solution to invertiblely protect face information in the full-process pipeline from camera to final users. Specifically, We design a novel lightweight Flow-based Face Encryption Method (FFEM) on the local embedded system privately connected to the camera, minimizing the risk of eavesdropping during data transmission. FFEM uses a flow-based face encoder to encode each face to a Gaussian distribution and encrypts the encoded face feature by random rotating the Gaussian distribution with the rotation matrix is as the password. While encrypted latent-variable face images are sent to users through public but less reliable channels, password will be protected through more secure channels through technologies such as asymmetric encryption, blockchain, or other sophisticated security schemes. User could select to decode an image with fake faces from the encrypted image on the public channel. Only trusted users are able to recover the original face using the encrypted matrix transmitted in secure channel. More interestingly, by tuning Gaussian ball in latent space, we could control the fairness of the replaced face on attributes such as gender and race. Extensive experiments demonstrate that our solution could protect privacy and enhance fairness with minimal effect on high-level downstream task.

1 Introduction

The prevalence of high-resolution cameras has generated unprecedented scale visual data, thus significantly stimulating computer vision applications. While promoting the common good, *i.e.*, autonomous driving (Yurtsever et al. 2020), artificial intelligence (AI) has also raised ethical concerns. For example, the leakage of privacy-sensitive information such as faces, car plate number poses the threat of social media profiles identification and tracking of user relations through large-scale deep face recognition (FR) analysis.

Being aware of the negative aspects of AI on the society, global legislative bodies actively call for actions to use AI for the public good of humanity as well as ensuring global sustainability set forth in the United Nation's Sustainable Development Goals (SDGs). For example, European Commission requires that all stakeholders partaking the AI system should ensure the respect for privacy and avoid unfair bias throughout the system's entire life cycle (AI 2019). To establish Society5.0, Japan also claims that AI should be developed, utilised and implemented following the principles of privacy protection and fairness (Secretariat et al. 2019).

In this avenue, we would like to discuss how to protect the privacy systematically for compliance with these regulations. Particularly, we focus on an important type of sensitive information-faces, that co-occurred with other objects captured by cameras. As shown in Figure 1, the earliest and most straightforward solution is directly obfuscating sensitive information by pixel-level processing such as blur, pixelation or adding Gaussian noise. While effective, these obfuscation methods either result in poor visual perception or leave negative efforts against consequent recognition algorithms (Cao et al. 2021). Generative methods offer an appealing way to replace the privacy information with more realistic image. On the other hand, methods like adversarial methods (Yang et al. 2021b) and differential privacy method (Chamikara et al. 2020) to apply perturbation to evade the recognition of a FR system.

Considering that when sharing photo with family or close friends, it is necessary to allow user to acquire the original images. To achieve this, an invertible face de-identification algorithm (Cao et al. 2021) is recently proposed to encrypt the images where the original original images could be restored with some passwords. Unfortunately, existing methods are so expensive that they can only be implemented on the cloud or on personal workstations. The uploading of captured original image to an encrypted site through public channels is exposed to the danger of being stolen by Bob, as illustrated in Figure 1. This eavesdropping danger also threats the password of (Cao et al. 2021).

In this article, we would like to share a systematic solution that protects the privacy in the entire lifecycle, from camera to end user. As shown in Figure 2, we choose to en-

*corresponding author

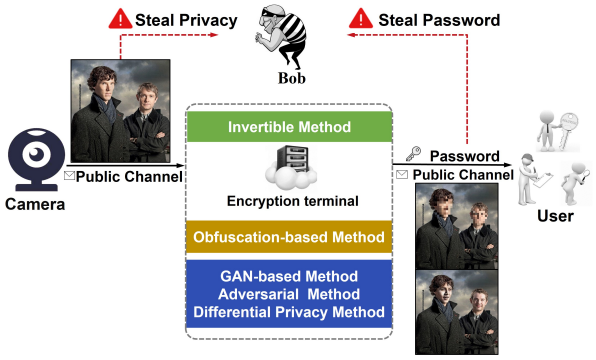

Figure 1: Existing privacy protection algorithms.

crypt the image immediately in local embedded system associated with cameras before uploading to intermediate or final users. This fundamentally avoids the privacy leakage on the transmission channel which is vulnerable to privacy thieves (Bob in Figure 1). As shown in Figure 2, we propose a Flow-based Face Encryption Method (FFEM) and implement it on the privacy encryption camera terminal. FFEM includes two steps: 1) detect, frontalize face and encode it to a Guassian distribution as encoded face representation using a flow-based (Kingma and Dhariwal 2018) encryption model; 2) rotate the encoded faces (Gaussian distribution) by an encryption matrix (orthogonal rotation matrix). Our FFEM is a low cost and lightweight model that runs on NVIDIA Jetson Nano with 4G memory size. Since the flow-based face encoding and decoding model are invertible, the original faces can also be recovered easily if the password is available (the rotation matrix in the encoded face latent variable spaces).

After local encryption, the privacy encryption camera terminal publishes two kinds of data: a private encrypted image and face posture information through a public channel and a rotation matrix password through a (virtual) secure channel. As illustrated in Figure 2, the encrypted latent-variable face image $\mathbf{I}_{enc}$ per se is equivalent to the obfuscated image. We could safely transmit this encrypted latent-variable face image and face posture in a public but less reliable channel. Users who are tolerant to obfuscation, like the one who use like privacy-enhanced version of LSVRC (Yang et al. 2022), could directly use $\mathbf{I}_{enc}$ for their recognition tasks. Benchmarking existing obfuscated datasets, we demonstrate that our encryption has minimal impact on downstream recognition models. For better visual perception, privacy- aimed users can also choose to obtain a de-identified image by decoding the encrypted latent-variable face image and posture information without a password, resulting in a fake face with the same pose as the original face. Interestingly, when we encrypt faces through a uniformly sampled orthogonal matrix(encryption matrix), each face will get a randomly encrypted face, which greatly promotes the fairness of attributes such as gender and race. By manipulating the Gaussian ball of encoded face representation, we could even control the degree of the fairness. We can tune the size of Gaussian ball to enforce that the decoded faces are closed to a mean face to guarantee the fairness. Thus, the overall fairness of the artificial intelligence system could be enhanced (Wang et al. 2022; Karkkainen and Joo 2021) when we replace the original faces with fake faces. Finally, for those trusted users or face-aimded user who target the raw face information, they can decrypt the or original image with the corresponding password. In our design, this password could be transmitted more reliably in the virtual secure channel, which can be achieved by asymmetric encryption, user authentication in blockchain or other sophisticated security protection schemes.

In this paper, we have made the following contributions:

- We propose systematic solution to protect the privacy from the camera to the end users. Specifically, our proposed flow-based face encryption method (FFEM) is low cost and lightweight that could be implemented on the local embedded system of privacy encryption camera terminal, minimizing the eavesdropping risk when transmitting data to encryption site by existing methods.
- By encoding the face to a latent variable in a Gaussian distribution (gassian ball) with a flow-based encoder, we could flexibly generate a encrypted latent-variable face by rotating the Gaussian distributed latent variable using the encryption matrix and generate a fake face using flow-based decoder. Only privacy-aimed user could decipher the original face with this encryption matrix transmitted through secure channel, while other users could choose to decode a nature but fake face from data transmitted in public channel.
- By manipulating the size of Gaussian ball in the latent space of FFEM, we could even control the fairness of the decoded fake images, thus enhancing the fairness in AI applications. Reduce the size of Gaussian ball can increase the fairness of encrypted faces by generating faces similar to mean face.
- Extensive experiments demonstrate our solution could effectively protect the privacy and enhance fairness with minimal negative effects on high-level tasks.

2 Related Work

Privacy Protection. The earliest face privacy protection algorithms are obfuscation-based methods, which obfuscate of face regions by occlusion, blurring, pixelation and etc (McPherson, Shokri, and Shmatikov 2016; Yang et al. 2022). However, these methods are now challenged by some obfuscation adapted face recognition (FR) systems (McPherson, Shokri, and Shmatikov 2016) and blurred or pixelation recovery methods (Gharbi et al. 2016; Yang et al. 2021a).

Since Deep neural networks are susceptible to adversarial examples and output incorrect results (Szegedy et al. 2013; Pang et al. 2020), the adversarial examples have been used to resist attacks by malicious face recognition systems (Yang et al. 2021b; Dong et al. 2018). However, most of these method are “white-box” attack that requires to know the FR system in advance, restricting their application on “black-box” FR system in real-life. Differential privacy (Dwork

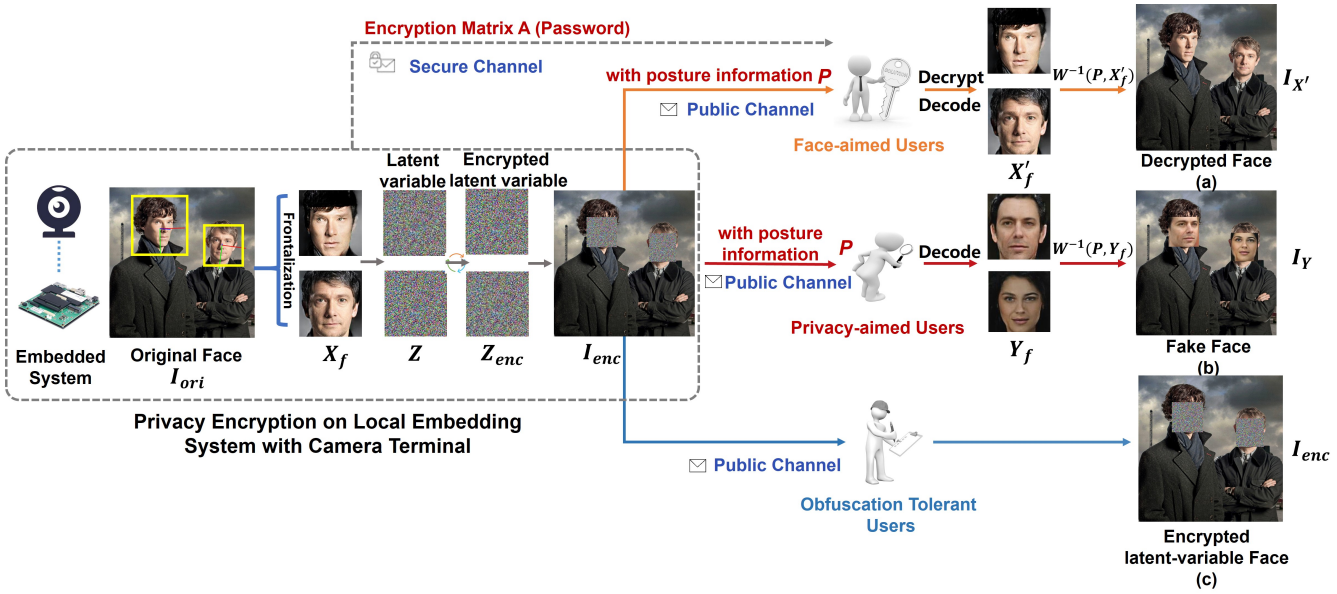

Figure 2: Assuming that our privacy encryption camera terminal captures some faces, it will convert them into encrypted latent-variable face and transmit them to different users through the public channel. Privacy encryption camera terminal will transmit the encrypted matrix to Face-aimed User through a secure channel so that Face-aimed User can decrypt the faces, while the other two types of users will use the encrypted latent-variable face or fake face to complete their tasks.

2008) is anther popular privacy protection strategy that removes individual features to protect user privacy while preserving statistical features (Cangialosi et al. 2022). The added Laplacian noise occasionally could not effectively destroy the privacy information, thus failing to protect against FR systems.

Fairness in AI. Many efforts are proposed to measure and enhance the fairness in AI. For example, IBM's AI Fairness 360 (Bellamy et al. 2019) measures the bias of a machine learning model to adjust the model's training set, ultimately promoting AI fairness. While more and more face recognition algorithms are used in everyday life, many of them have much higher false positive rates for non-white faces than white faces, which would affect judicial fairness (Salvador et al. 2021). To increase the fairness of the model, one effective solution is to process the dataset for the model training, thus avoiding an unfair model is derived from an unfair dataset (Karkkainen and Joo 2021). Another approach is to retrain the model to achieve fairness (Gong, Liu, and Jain 2020), but it is computationally expensive and may reveal private information about the data used for retraining.

Generative Model. Generative model can be divided into three types: (1) Generative adversarial networks (GANs) (Goodfellow et al. 2014), which plays a max-min game until generate data distribution is similar to the real data. (2) Variational autoencoders (VAEs) (Kingma and Ba 2014) represent high-dimensional complex data by learning a low-dimensional latent space in an unsupervised manner. (3) Flow-based model was first proposed in NICE (Dinh, Krueger, and Bengio 2014) and extended in RealNVP (Dinh, Sohl-Dickstein, and Bengio 2016). They use a flow containing the equivalent of a permutation that reverses the ordering of the channels. Glow (Kingma and Dhariwal 2018) replaces this fixed permutation with a (learned) invertible 1 × 1 convolution, where the weight matrix is initialized as a random rotation matrix. The simplification of calculation of the matrix significantly reduces the overall computational complexity.

The flow-based model maps the sampling space to an intermediate explicit representation by training an invertible transformation, allowing us to recover the original data directly from this intermediate representation. Due to its invertible property and low computational complexity, we select flow-based model as our generative model.

3 Privacy Encryption on Camera Terminal

Our framework of our method is shown in Figure 2. On the privacy encryption camera terminal, we implement our Flow-based Face Encryption Method (FFEM) to generate privacy-protected images.

3.1 Face Detection and Rotation

Face Detection. The first step detects and resizes face $\mathbf{X} \in \mathbf{R}^{3\times m\times m}$ in the image captured by the camera. m is the height and width of the cropped face image. In our system, we choose Yolo5Face (Qi et al. 2021) for face detection due to the balance between accuracy and computational complexity.

Pose Estimation and Frontalization. Since training a flow-based model handling multiple face poses requires much parameters, the model size would be too large for an embedded system. To reduce the computational and memory

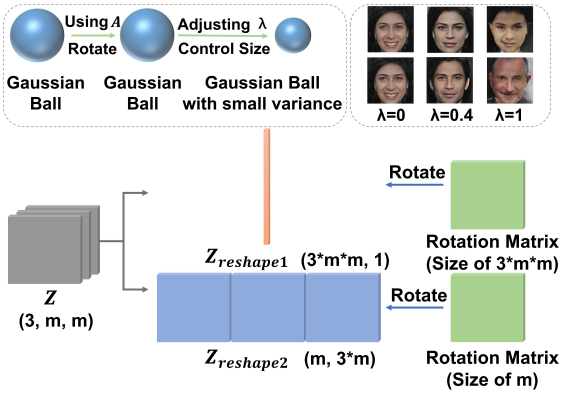

Figure 3: Illustration of encryption matrices of different sizes.

requirement, we train the flow-based model only for single frontal pose. Therefore, before encryption, we frontalize the face and also record the face pose $\mathbf{P} = (yaw, pitch, roll)$ which contains the angles of three Euler angles(yaw, pitch, roll) for the consequent recovery. Here, we use lightweight 3ddfa (Guo et al. 2020) to estimate the pose estimation. For frontalization, we choose Rotate and Render (Zhou et al. 2020) to wrap the original face $\mathbf{X}$ to the frontal face $\mathbf{X}_f$: $\mathbf{X}_f = \mathbf{W}(\mathbf{P}, \mathbf{X})$, where $\mathbf{W}(\cdot)$ is the pose wrapping function.

3.2 Flow-Based Face Encryption

We propose a flow-based method to encrypt the original face image $\mathbf{I}_{ori}$ by mapping its data distribution to a Gaussian distribution $\mathcal{H} \triangleq \mathcal{N}(0, I_m)$ in a latent space. Since the shape of the Gaussian distribution $\mathcal{H}$ in three-dimensional space is is a ball, we name it as Gaussian ball.

At first, we train an flow-based model that learns the bijection mapping between the face in image space and the latent variable space. NICE (Dinh, Krueger, and Bengio 2014) and RealNVP (Dinh, Sohl-Dickstein, and Bengio 2016) use a flow containing the equivalent of a permutation that reverses the ordering of the channels. Glow replace this fixed permutation with a (learned) invertible 1×1 convolution. We adopt the flow-based model of Glow (Kingma and Dhariwal 2018) in our method. Since the flow-based model is reversible, we use the forward channel of the flow-based model as the encoder and the reverse channel as the decoder. The encoder $f(\cdot)$: $\mathbf{Z} = f(\mathbf{X}_f)$ maps the original frontal face $\mathbf{X}_f$ to the latent space $\mathbf{Z} \in \mathbf{R}^{3\times m\times m}, \mathbf{Z} \sim \mathcal{N}(0, \mathbf{I})$. Since the distribution of the latent variables is on a Gaussian ball, the rotation of the latent variables will not change the distribution of the latent variables (Wu, Du, and Yuan 2020). Thus, we rotate encrypt this latent variable $\mathbf{Z}$ with a uniformly sampled orthogonal matrix which is called "encryption matrix" $\mathbf{A}$: $\mathbf{Z}_{enc} = \mathbf{A} \cdot \mathbf{Z}$. Finally, we cover the encrypted $\mathbf{Z}_{enc}$ where the face is in the original image. By reshaping the latent variable $\mathbf{Z} \in \mathbf{R}^{3\times m\times m}$ into $\mathbf{R}^{m\times 3m}$. The size of encryption matrix $\mathbf{A}$ is only $m \times m$, which is 128Kb for an image sized 128×128. An illustration of the different encryption matrices is shown in Figure 3.

Figure 4: Effect of change of temperature. From left to right, the temperature is 0, 0.2, 0.4, 0.6, 0.8, 1.0

Now we have two kinds of data: 1) privacy-protected data: encrypted latent-variable face image $\mathbf{I}_{enc}$ and posture information $\mathbf{P}$. 2) privacy-sensitive data: Encryption matrix $\mathbf{A}$. As discussed in next section, the former are transmitted on public but less reliable channel while the latter goes through secure channel.

3.3 Fairness Enhancement

Multiplying each latent variable $\mathbf{Z} \sim \mathcal{N}(0, \lambda^2\mathbf{I})$ with a temperature $\lambda \in R$ could control the variance of its Gaussian distribution, thus manipulating the size of the Gaussian ball. As shown in Figure 4, we can see that as λ decreases, the ages of two encrypted faces become closer and more feminine. Therefore, by reducing λ, the bias caused by gender and age can be reduced to promote fairness. Using a smaller λ will compress the size of the Gaussian ball, making the generated face closer to the "average face" learned by the model, reducing the diversity between different generated faces. So we can control the quality and diversity of the faces generated by the encoder by adjusting the λ, we set the λ to 0.5 to obtain a better generation quality and reduce the diversity of faces. The experiment shows that our method could also promote racial fairness.

4 Data Transmission and End User

Posture Restoration. Since the rotation angle of Rotate and Render only includes yaw and pitch, the rotation of roll cannot be achieved. We choose CFR-GAN (Ju et al. 2022) to render the face with same pose original image. Both of the methods we have chosen to frontalization and posture restoration require neither human intervention nor paired data.

Different Users. For the end users, they will perform different operations on latent-variable face image $\mathbf{I}_{enc}$ transmitted over public channel. Because its face privacy information has been effectively protected, it can be transmitted through insecure public channels. In the following, we will introduce these three different types of users and their operation process on the user server.

4.1 Face-Aimed User

The face-aimed user deciphers the latent-variable face in encrypted latent-variable face image $\mathbf{I}_{enc}$ to a deciphered latent variable using the encryption matrix transmitted through the secure channel: $\mathbf{Z}^{'} = \mathbf{A}^{-1} \cdot \mathbf{Z}_{enc}$.

The encryption matrix is transmitted over a secure channel, so channel eavesdropping can be prevented. When they

Model	Backbone	Easy	Medium	Hard	FLOPs(G)
DSFD (Li et al. 2019)	ResNet152	94.29%	91.47%	71.39%	259.55
RetinaFace (Deng et al. 2020)	ResNet50	94.92%	91.90%	64.17%	37.59
HAMBox (Liu et al. 2019)	ResNet50	95.27%	93.76%	76.75%	43.28
RetinaFace (small)	MobileNet0.25 (Howard et al. 2017)	87.78%	81.16%	47.32%	0.802
FaceBoxes (Zhang et al. 2017)	FaceBoxes	76.17%	57.17%	24.18%	0.275
YOLOv5s	YOLOv5-CSPNet (Jocher 2020)	94.67%	92.75%	83.03%	5.751
YOLOv5n	ShuffleNetv2 (Ma et al. 2018)	93.74%	91.54%	80.32%	2.111

Table 1: Comparison of Yolo5Face and existing face detectors on the Widerface validation dataset.

obtain the decrypted latent variable, they decode it to a decrypted frontal face $\mathbf{X}_f^{'}$ through the inverse channel of the flow model: $\mathbf{X}_f^{'} = f^{-1}(\mathbf{Z}^{'})$.

After obtaining the decrypted frontal face, they use the posture information transmitted over public channel to recover the decrypted face to the original pose $\mathbf{X}^{'}$: $\mathbf{X}^{'} = \mathbf{W}^{-1}(\mathbf{P}, \mathbf{X}_f^{'})$. The decrypted face image $\mathbf{I}_{X'}$ is shown in Figure 2 (a).

4.2 Privacy-Aimed User

Because the flow-based model is invertible, the user could use the decoder $f^{-1}(\cdot)$ to decode a fake face. The privacy-aimed users decode the encrypted latent-variable face to a encrypted frontal face $\mathbf{Y}_f$ through the inverse channel of the flow model: $\mathbf{Y}_f = f^{-1}(\mathbf{Z}_{enc})$.

Then they use the posture information P transmitted over public channel to rotate the encrypted frontal face to the original pose to reconstruct the fake face $\mathbf{Y}$: $\mathbf{Y} = \mathbf{W}^{-1}(\mathbf{P}, \mathbf{Y}_f)$. The fake face image $\mathbf{I}_Y$ is shown in Figure 2 (b). We define this encryption method as fake face method.

4.3 Obfuscation Tolerant User

There are some users which we call obfuscation tolerant users, do not need face information. Similar to the user of privacy-enhanced version of ILSRC (Yang et al. 2022), they can directly use the encrypted latent-variable face image $\mathbf{I}_{enc}$ for their downstream tasks as shown in Figure 2 (c). We define this encryption method as encrypted latent-variable face method.

5 Experiments

We at first train our FFEM on images of CelebA-HQ (Karras et al. 2017) and FFHQ (Karras, Laine, and Aila 2019) dataset. We at first detect the faces of these 100k images and frontalize the faces through Rotate and Render (Zhou et al. 2020) and use 90% of them for training and rest 10% for testing. The batch size, temperature, learning rate, levels, steps is set to 16, 0.5, 0.001, 5 and 32 respectively. As we can see, the fake faces decoded by privacy-aimed user still visually maintain the appearance of human faces, showing our FFEM captures the semantics of the input face images.

We then demonstrate our performance on privacy protection and fairness enhancement with following three experiments. They are performed on two NVIDIA RTX 3090.

(a) Original Faces

(b) Fake Faces

Figure 5: Visualization of the faces before and after encryption. The first row contains original faces and the second row contains the corresponding fake faces.

5.1 Dataset

CelebA-HQ CelebA-HQ dataset is a high resolution version of CelebA dataset (Liu et al. 2015). It includes a total of 30k high-resolution celebrity faces.

FFHQ FFHQ dataset is a high-quality face dataset containing 70k high-definition face images with a resolution of 1024×1024, which are diverse and distinct in age, genders, races, skin colors, expressions, face shapes, hairstyles, facial postures and image background. For both datasets, we use its 256x256 resolution version and resize its images to 128x128 after face detection.

LFW LFW dataset (Huang and Learned-Miller 2014) mainly collects images from the internet including more than 13k face images in total. Each image is identified with the name of the corresponding person.

UTK-face UTK-face dataset (Zhang, Song, and Qi 2017) is a dataset with annotated race, age, and gender. Its age ranges from 0 to 116 years old. Gender is divided into male and female. Races are divided into five categories: White, Black, Asian, Indian, and Others.

HMDB51 HMDB51 dataset (Kuehne et al. 2011) contains 51 categories of actions, with a total of 6849 videos collected from YouTube, google videos,*etc.* Each action contains at least 51 videos and the video resolution is 320x240.

5.2 Effectiveness of Face Encryption

We evaluate the privacy protection against four widely used third-party "black-box" FR models: ArcFace (Deng et al. 2019), CosFace (Wang et al. 2018), FaceNet (Schroff, Kalenichenko, and Philbin 2015), and SphereFace (Liu et al. 2017).

Similar to (Yang et al. 2021b), we randomly select 500 faces with different identities (containing two or more faces)

in the LFW dataset as the probe images, leaving all of the remaining faces as the gallery. FR model calculates the similarity between the probe image $\boldsymbol{x}$ and each face in the gallery. We report Top-1 and Top-5 Accuracy representing 1 or 5 faces with the highest similarity have faces with the same identity as $\boldsymbol{x}$. The lower Top-1 and Top-5 accuracy means better privacy protection. In this experiment, we evaluate on our encrypted latent-variable face method and fake face method. The encryption effect of the fake face method is shown in Figure 5.

Table 2 shows FR models could effectively recognize the identity of the faces before encryption, the face recognition models can recognize the identity of the face. After our encryption, these FR models fail to make identification. Compared with other face privacy protection methods, top-1 and top-5 accuracy on our encrypted latent-variable faces and fake faces achieved the lowest recognition accuracy among most FR systems. We attribute the stronger protection to adding information about new faces to confuse FR model. While DeepPrivacy (Hukkelås, Mester, and Lindseth 2019) replaces the original face with a new one, it still retains some original information, deteriorating the privacy protection ability. Experiments demonstrate that our encryption effectively protect the private information.

5.3 Effect on Downstream High-Level Task

We perform a video action recognition task to evaluate how our encryption affects the downstream high-level tasks. First, we pretrain a ResNet-50 (He et al. 2016) model on Kinetics-700 (Carreira et al. 2019) and Moments in Time (Monfort et al. 2019), and then finetune on the official training set of HMDB51 (Kuehne et al. 2011). After that, we preprocess the video frames of the validation set of HMDB51(divided into three splits), sample the video frames and the number of sampled frames per video does not exceed 100 frames. In the sampled frames, we only encrypt the faces larger than 24x24, because smaller ones are hard to recognize already. Finally we compare the Top-N accuracy of the action recognition task before and after encryption. Since our encrypted latent-variable face and fake face affect the facial expression, we do not evaluate on the categories such as smile and laugh.

Table 3 shows that, similar to pixelation, encrypted latent-variable face has little effect on action detection recognition and even slightly improves the accuracy. We suspect both pixelation and encrypted latent-variable face make the action recognition model to focus more on other regions unrelated to the face, potentially allowing it to make more accurate judgments. Compared to the accuracy before encryption, the accuracy of fake face encryption method drops slightly (0.13%-0.92%). The experiment demonstrate that both obfuscation tolerant users and privacy-aimed users could successfully perform the downstream recognition on our encryption methods.

5.4 Fairness Enhancement

Here, we verify how our method enhance the fairness on face detection experiment. Because the fairness experiment requires face information, we adopt the **fake face method**. We at first collect the Ground Truth(GT) with bbox of the faces detected by Yolo5Face (Qi et al. 2021) on UTK-face. We evaluate face detection model of OpenCV (Bradski and Kaehler 2008) on the fake face images and report the IoU. Finally, we use AI fairness 360 to calculate the Mean Difference(closer to zero implys greater fairness) and Disparate Impact(closer to 100% implys greater fairness) of different groups for fairness measurement.

According to the face detection results of OpenCV, we found that the average IoU of male is higher than that of female, and the average IoU of Indians is higher than that of other races. Therefore, in gender group, we define male as the privileged group, and female as the unprivileged one. For racial group, we define Indian as a privileged group and others as the unprivileged ones. In Table 4, the Mean Difference between the unprivileged group and the privileged group after encryption is closer to 0% compared to original ones, illustrating a significant more fair performance. Disparate Impact is also closer to 100%, which shows that gender differences and racial differences are much smaller on our fake faces. We have also evaluate the fairness improvement on individual races and find our method achieves a reduction in overall bias. Please refer to the supplementary for details.

In order to verify that our method is effective in eliminating the bias of the machine's first impression of a person, we apply the fairness performance on a commercial face attractiveness rating API available at Face++ Platform[1] to evaluate the attractiveness of original and fake faces on a subset of the UTK-face dataset (containing 3250 faces).

From Table 5 shows attractiveness score differs obvious on gender in original images with the mean difference between genders 14.3%. But after encryption, the mean difference between genders dropped to 7.73%, achieving better fairness. Disparate Impact also dropped from 133.55% to 115.87%, indicating that the effects of different genders on the results are more balanced. The experiment, reported in supplementary, on race also shows a significant overall fairness enhancement through our method.

Controlling fairness with parameters λ. As discussed in Section 3.3, the fairness enhancement could be adjusted by λ. We evaluate on the gender groups in UTK-Face subset. Table 6 shows that a smaller λ results in higher fairness, represented by lower mean difference on gender groups. Meanwhile, decreasing λ also decreases the standard deviation(STD) of the attractiveness score lower, showing smaller difference between each individual group.

5.5 Embedded Devices Deployment

As the proof of concept system, we set up an embedded system using NVIDIA Jetson Nano, a popular IoT device as the privacy encryption camera terminal to deploy the FFEM. NVIDIA Jetson Nano integrates a Quad-core ARM processor, a 128 NVIDIA CUDA core GPU, and 4GB unified memory [2]. To reduce the memory usage and optimize the inference performance, we deploy our model with

[1] https://www.faceplusplus.com.cn

[2] https://developer.nvidia.com/embedded/jetson-nano

	FaceNet		CosFace		ArcFace		SphereFace	
	top-1	top-5	top-1	top-5	top-1	top-5	top-1	top-5
Original	93.4%	97.8%	95.4%	97.8%	96.0%	98.2%	89.4%	93.4%
Blacked out	0.4%	1.6%	0%	0.8%	0.4%	1.6%	0.2%	1.4%
Pixelation(8x8)	0.2%	0.6%	0.2%	1.2%	0.2%	1.2%	0%	1.4%
Pixelation(16x16)	0%	0.4%	0%	1.0%	0.4%	1.4%	0.2%	1.0%
Gaussian Blur(3x3)	0.2%	0.8%	0.2%	0.8%	0.2%	0.4%	0.2%	1.0%
Gaussian Blur(9x9)	0.4%	1.6%	0.6%	1.2%	0.2%	1.2%	0.2%	1.6%
Heavy Blur	0.2%	1.0%	0.2%	1.2%	0.2%	0.6%	0%	1.0%
DeepPrivacy	0%	1.0%	0.2%	1.2%	0.2%	0.6%	0%	1.0%
Encrypted latent-variable face	0%	**0%**	0%	0.6%	0.4%	1.0%	0%	1.2%
Fake face	**0%**	0.2%	**0%**	**0%**	**0%**	**0%**	**0%**	**0%**

Table 2: Comparison of different face privacy protection methods.

	Split 1		Split 2		Split 3	
	top-1	top-5	top-1	top-5	top-1	top-5
Original	61.70%	89.87%	83.27%	94.90%	82.35%	95.75%
Pixelation	61.63%	89.73%	83.33%	95.10%	82.55%	95.88%
Encrypted latent-variable face	61.70%	89.80%	83.27%	95.03%	82.29%	95.82%
Fake face	61.31%	89.28%	82.35%	94.77%	81.70%	95.36%

Table 3: Top-N accuracy of the action recognition task between different methods.

	Face	Mean Difference	Disparate Impact
Gender	Original	-7.31%	91.17%
	Encrypted	**-2.60%**	**97.03%**
Race	Original	-3.63%	95.20%
	Encrypted	**1.19%**	**101.43%**

Table 4: Mean Difference and Disparate Impact between unprivileged and privileged groups before and after encryption.

	Face	Mean Difference	Disparate Impact
Gender	Original	14.3%	133.55%
	Encrypted	**7.73%**	**115.87%**

Table 5: Mean Difference and Disparate Impact between Female and male before and after encryption.

	0.5	0.6	0.7	0.8
Mean Difference	7.73%	10.8%	12.7%	12.51%
STD	10.25	11.76	12.41	12.37

Table 6: Fairness results under different λ.

Activities	**Time (ms)**	**Energy (J)**		
		GPU	CPU	Board
Encoding	267	0.24	0.53	1.2
Encoding + Rotation	307	0.29	0.62	1.3

Table 7: Privacy Encryption Time and Energy Cost on Embedded Device.

TensorRT [3](a C++ library that facilitates high performance inference on NVIDIA GPUs), on the embedded system. The software configuration of our experimental platfrom includes CUDA version 10.2, python 3.6.9 and TensorRT 8.0.1.6. To evaluate the time and energy cost of privacy encryption phase, we conducted experiments on 100 images and obtained the averaged results. in . According to Table 7, it takes around 300 ms and 1.3 Joule to encrypt an image on the embedded device.

6 Discussion and Limitation

Compliance with legitimate efforts on protecting the privacy and boosting the fairness in the entitle life cycle, we have proposed a systematic solution from cameras to users. Different from existing research focusing on improving the algorithms, we stood on the shoulders of giants to address less considered problems, such as the dangers of eavesdropping on data transmissions. Extensive experiments demonstrate that our solution can effectively protect the privacy, enhance the fairness while imposing limited negative effects on the down-stream high level tasks for different kinds of users. The proof-of-concept evaluation of the embedded system demonstrated speed and energy efficiency, showing great potential for widespread deployment of the solution. While effective in terms of privacy protection, this initial attempt leaves much to be desired. For example, the decoded face, although rotated, still showed an unnatural background when merged into the image. On this path, we hope to engage in discussions with experts from different fields to find out more practical applications in real-world systems.

[3]https://developer.nvidia.com/tensorrt

Acknowledgments

This work was supported by the Guangdong Science Fund Grant Number 2021A1515011915 and JST Moonshot R&D Grant Number JPMJMS2011 and JST ACT-X Grant Number JPMJAX190D, Japan.

References

AI, H. 2019. High-level expert group on artificial intelligence. *Ethics guidelines for trustworthy AI*, 6.

Bellamy, R. K.; Dey, K.; Hind, M.; Hoffman, S. C.; Houde, S.; Kannan, K.; Lohia, P.; Martino, J.; Mehta, S.; Mojsilović, A.; et al. 2019. AI Fairness 360: An extensible toolkit for detecting and mitigating algorithmic bias. *IBM Journal of Research and Development*, 63(4/5): 4–1.

Bradski, G.; and Kaehler, A. 2008. *Learning OpenCV: Computer vision with the OpenCV library*. " O'Reilly Media, Inc.".

Cangialosi, F.; Agarwal, N.; Arun, V.; Narayana, S.; Sarwate, A.; and Netravali, R. 2022. Privid: Practical,{Privacy-Preserving} Video Analytics Queries. In *19th USENIX Symposium on Networked Systems Design and Implementation (NSDI 22)*, 209–228.

Cao, J.; Liu, B.; Wen, Y.; Xie, R.; and Song, L. 2021. Personalized and Invertible Face De-identification by Disentangled Identity Information Manipulation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 3334–3342.

Carreira, J.; Noland, E.; Hillier, C.; and Zisserman, A. 2019. A short note on the kinetics-700 human action dataset. *arXiv preprint arXiv:1907.06987*.

Chamikara, M. A. P.; Bertok, P.; Khalil, I.; Liu, D.; and Camtepe, S. 2020. Privacy preserving face recognition utilizing differential privacy. *Computers & Security*, 97: 101951.

Deng, J.; Guo, J.; Ververas, E.; Kotsia, I.; and Zafeiriou, S. 2020. Retinaface: Single-shot multi-level face localisation in the wild. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 5203–5212.

Deng, J.; Guo, J.; Xue, N.; and Zafeiriou, S. 2019. Arcface: Additive angular margin loss for deep face recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 4690–4699.

Dinh, L.; Krueger, D.; and Bengio, Y. 2014. Nice: Nonlinear independent components estimation. *arXiv preprint arXiv:1410.8516*.

Dinh, L.; Sohl-Dickstein, J.; and Bengio, S. 2016. Density estimation using real nvp. *arXiv preprint arXiv:1605.08803*.

Dong, Y.; Liao, F.; Pang, T.; Su, H.; Zhu, J.; Hu, X.; and Li, J. 2018. Boosting adversarial attacks with momentum. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 9185–9193.

Dwork, C. 2008. Differential privacy: A survey of results. In *International conference on theory and applications of models of computation*, 1–19. Springer.

Gharbi, M.; Chaurasia, G.; Paris, S.; and Durand, F. 2016. Deep joint demosaicking and denoising. *ACM Transactions on Graphics (ToG)*, 35(6): 1–12.

Gong, S.; Liu, X.; and Jain, A. K. 2020. Mitigating Face Recognition Bias via Group Adaptive Classifier. *arXiv e-prints*.

Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; and Bengio, Y. 2014. Generative adversarial nets. *Advances in neural information processing systems*, 27.

Guo, J.; Zhu, X.; Yang, Y.; Yang, F.; Lei, Z.; and Li, S. Z. 2020. Towards Fast, Accurate and Stable 3D Dense Face Alignment. In *Proceedings of the European Conference on Computer Vision (ECCV)*.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 770–778.

Howard, A. G.; Zhu, M.; Chen, B.; Kalenichenko, D.; Wang, W.; Weyand, T.; Andreetto, M.; and Adam, H. 2017. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*.

Huang, G. B.; and Learned-Miller, E. 2014. Labeled faces in the wild: Updates and new reporting procedures. *Dept. Comput. Sci., Univ. Massachusetts Amherst, Amherst, MA, USA, Tech. Rep*, 14(003).

Hukkelås, H.; Mester, R.; and Lindseth, F. 2019. Deepprivacy: A generative adversarial network for face anonymization. In *International symposium on visual computing*, 565–578. Springer.

Jocher, G. 2020. Yolov5. https://github.com/ultralytics/yolov5. Accessed: 2020-05-29.

Ju, Y.-J.; Lee, G.-H.; Hong, J.-H.; and Lee, S.-W. 2022. Complete face recovery gan: Unsupervised joint face rotation and de-occlusion from a single-view image. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 3711–3721.

Karkkainen, K.; and Joo, J. 2021. Fairface: Face attribute dataset for balanced race, gender, and age for bias measurement and mitigation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 1548–1558.

Karras, T.; Aila, T.; Laine, S.; and Lehtinen, J. 2017. Progressive growing of gans for improved quality, stability, and variation. *arXiv preprint arXiv:1710.10196*.

Karras, T.; Laine, S.; and Aila, T. 2019. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 4401–4410.

Kingma, D. P.; and Ba, J. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Kingma, D. P.; and Dhariwal, P. 2018. Glow: Generative flow with invertible 1x1 convolutions. *Advances in neural information processing systems*, 31.

Kuehne, H.; Jhuang, H.; Garrote, E.; Poggio, T.; and Serre, T. 2011. HMDB: a large video database for human motion

recognition. In *Proceedings of the International Conference on Computer Vision (ICCV)*.

Li, J.; Wang, Y.; Wang, C.; Tai, Y.; Qian, J.; Yang, J.; Wang, C.; Li, J.; and Huang, F. 2019. DSFD: dual shot face detector. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 5060–5069.

Liu, W.; Wen, Y.; Yu, Z.; Li, M.; Raj, B.; and Song, L. 2017. Sphereface: Deep hypersphere embedding for face recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 212–220.

Liu, Y.; Tang, X.; Wu, X.; Han, J.; Liu, J.; and Ding, E. 2019. Hambox: Delving into online high-quality anchors mining for detecting outer faces. *arXiv preprint arXiv:1912.09231*.

Liu, Z.; Luo, P.; Wang, X.; and Tang, X. 2015. Deep learning face attributes in the wild. In *Proceedings of the IEEE international conference on computer vision*, 3730–3738.

Ma, N.; Zhang, X.; Zheng, H.-T.; and Sun, J. 2018. Shufflenet v2: Practical guidelines for efficient cnn architecture design. In *Proceedings of the European conference on computer vision (ECCV)*, 116–131.

McPherson, R.; Shokri, R.; and Shmatikov, V. 2016. Defeating image obfuscation with deep learning. *arXiv preprint arXiv:1609.00408*.

Monfort, M.; Andonian, A.; Zhou, B.; Ramakrishnan, K.; Bargal, S. A.; Yan, T.; Brown, L.; Fan, Q.; Gutfreund, D.; Vondrick, C.; et al. 2019. Moments in time dataset: one million videos for event understanding. *IEEE transactions on pattern analysis and machine intelligence*, 42(2): 502–508.

Pang, T.; Yang, X.; Dong, Y.; Xu, K.; Zhu, J.; and Su, H. 2020. Boosting adversarial training with hypersphere embedding. *Advances in Neural Information Processing Systems*, 33: 7779–7792.

Qi, D.; Tan, W.; Yao, Q.; and Liu, J. 2021. YOLO5Face: why reinventing a face detector. *arXiv preprint arXiv:2105.12931*.

Salvador, T.; Cairns, S.; Voleti, V.; Marshall, N.; and Oberman, A. 2021. FairCal: Fairness Calibration for Face Verification. *arXiv preprint arXiv:2106.03761*.

Schroff, F.; Kalenichenko, D.; and Philbin, J. 2015. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 815–823.

Secretariat, C.; et al. 2019. Social Principles of Human-Centric AI. Cabinet Secretariat.

Szegedy, C.; Zaremba, W.; Sutskever, I.; Bruna, J.; Erhan, D.; Goodfellow, I.; and Fergus, R. 2013. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*.

Wang, H.; Wang, Y.; Zhou, Z.; Ji, X.; Gong, D.; Zhou, J.; Li, Z.; and Liu, W. 2018. Cosface: Large margin cosine loss for deep face recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 5265–5274.

Wang, Z.; Dong, X.; Xue, H.; Zhang, Z.; Chiu, W.; Wei, T.; and Ren, K. 2022. Fairness-aware Adversarial Perturbation Towards Bias Mitigation for Deployed Deep Models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 10379–10388.

Wu, C.; Du, C.; and Yuan, Y. 2020. Secure Data Sharing With Flow Model. *arXiv preprint arXiv:2009.11762*.

Yang, K.; Yau, J. H.; Fei-Fei, L.; Deng, J.; and Russakovsky, O. 2022. A study of face obfuscation in imagenet. In *International Conference on Machine Learning*, 25313–25330. PMLR.

Yang, T.; Ren, P.; Xie, X.; and Zhang, L. 2021a. Gan prior embedded network for blind face restoration in the wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 672–681.

Yang, X.; Dong, Y.; Pang, T.; Su, H.; Zhu, J.; Chen, Y.; and Xue, H. 2021b. Towards face encryption by generating adversarial identity masks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 3897–3907.

Yurtsever, E.; Lambert, J.; Carballo, A.; and Takeda, K. 2020. A survey of autonomous driving: Common practices and emerging technologies. *IEEE access*, 8: 58443–58469.

Zhang, S.; Zhu, X.; Lei, Z.; Shi, H.; Wang, X.; and Li, S. Z. 2017. Faceboxes: A CPU real-time face detector with high accuracy. In *2017 IEEE International Joint Conference on Biometrics (IJCB)*, 1–9. IEEE.

Zhang, Z.; Song, Y.; and Qi, H. 2017. Age Progression/Regression by Conditional Adversarial Autoencoder. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE.

Zhou, H.; Liu, J.; Liu, Z.; Liu, Y.; and Wang, X. 2020. Rotate-and-render: Unsupervised photorealistic face rotation from single-view images. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 5911–5920.