Restoring Gaussian Blurred Face Images for Deanonymization Attacks

Haoyu Zhai*, Shuo Wang*, Pirouz Naghavi, Qingying Hao, Gang Wang
University of Illinois Urbana-Champaign
{*zhai11, shuow6, naghavi2, qhao2, gangw*}*@illinois.edu*

***Abstract*—Gaussian blur is widely used to blur human faces in sensitive photos before the photos are posted on the Internet. However, it is unclear to what extent the blurred faces can be restored and used to re-identify the person, especially under a *high-blurring* setting. In this paper, we explore this question by developing a deblurring method called `Revelio`. The key intuition is to leverage a generative model's memorization effect and approximate the inverse function of Gaussian blur for face restoration. Compared with existing methods, we design the deblurring process to be *identity-preserving*. It uses a conditional Diffusion model for preliminary face restoration and then uses an identity retrieval model to retrieve related images to further enhance fidelity. We evaluate `Revelio` with large public face image datasets and show that it can effectively restore blurred faces, especially under a high-blurring setting. It has a re-identification accuracy of 95.9%, outperforming existing solutions. The result suggests that Gaussian blur should not be used for face anonymization purposes. We also demonstrate the robustness of this method against mismatched Gaussian kernel sizes and functions, and test preliminary countermeasures and adaptive attacks to inspire future work.**

1. Introduction

With the rise of online social networks, search engines, and content-sharing platforms, billions of photos are circulating on the Internet, many of which contain *identifiable human faces*. For privacy considerations, users often *blur* the faces in a sensitive photo before posting it on the Internet. For example, news media may publish photos of crime scenes and blur the faces of the victims/offenders to protect their privacy [1], [2]. Similarly, people who post photos of civil unrest usually blur the faces of protesters [3], [4]. Social media users who post photos of themselves/friends/strangers may choose to blur the faces if the photos capture sensitive, unflattering, or even illegal activities (e.g., drinking, stealing, or using drugs).

Gaussian blur [5] is among the most commonly used blurring algorithms [6], [7] to smooth and remove high-frequency details in face images. This is done by weighted-averaging on every pixel with its neighboring pixels where the weights are based on a Gaussian distribution (realized by a convolution kernel). Under a high-blurring setting, Gaussian blur can make the face unidentifiable to human eyes and thus is often used as an anonymization tool. The wide use of Gaussian blur, especially by lay users, is largely due to its availability in everyday photo-processing software and apps such as Adobe Photoshop [8], Apple's Motion [9], Android's Blur Photo Editor [10], PicsArt [11], and ASPOSE [12].

Motivation. In this paper, we ask one basic question: *To what extent can adversaries restore a Gaussian blurred face, and identify the person in the photo*. We particularly explore this question under a *high-blurring* level to push the limit of the deblurring method. Prior works have explored related questions but under different deanonymization contexts. For example, Cavedon et al. [13] showed that *image pixelization* could be reversed using a Maximum A Posteriori (MAP) method but their method cannot be applied to Gaussian blur. In addition, their method is customized for video streams (i.e., requiring multiple video frames) instead of a single photo. Hill et al. [7] used a Hidden Markov Model (HMM) to restore *redacted text* on documents. This method relies on the fixed set of English characters and digits for text recovery, but does not apply to human faces.

More recently, related work from the machine learning community seeks to address the (blind) image restoration problem [14], [15], [16], [17], [18]. Their goal is to restore high-quality images from degraded photos without prior knowledge of the degradation process. These works *cannot answer our research questions* for two reasons. First, most existing solutions are designed to restore unintentional image blur caused by camera shakes, poor lighting conditions, and low-quality cameras. The target degradation strength is usually low. In contrast, we focus on Gaussian blur *intentionally applied* for privacy protection, and thus the expected blurring level is much higher. As shown in Figure 1, under a heavy-blur setting (with a Gaussian kernel size of 81), existing solutions have a subpar performance. Second and more importantly, existing solutions are not designed to be *identity-aware*. They focus on re-generating facial details but the face does not need to be that of the original person/identity (Figure 1). In contrast, we want to preserve the identity of the original image. Note that, one recent solution Fantômas [18] indeed targeted face anonymization scenarios. However, they also did not explore restoring faces from highly blurred images (their kernel size is only 29). We re-trained their method under a heavy-blur setting, and Figure 1 shows that the face restoration is still suboptimal.

Our Approach. To fill in the gap and answer our research questions, we develop a system called `Revelio`.

*. These authors contributed equally to this work.

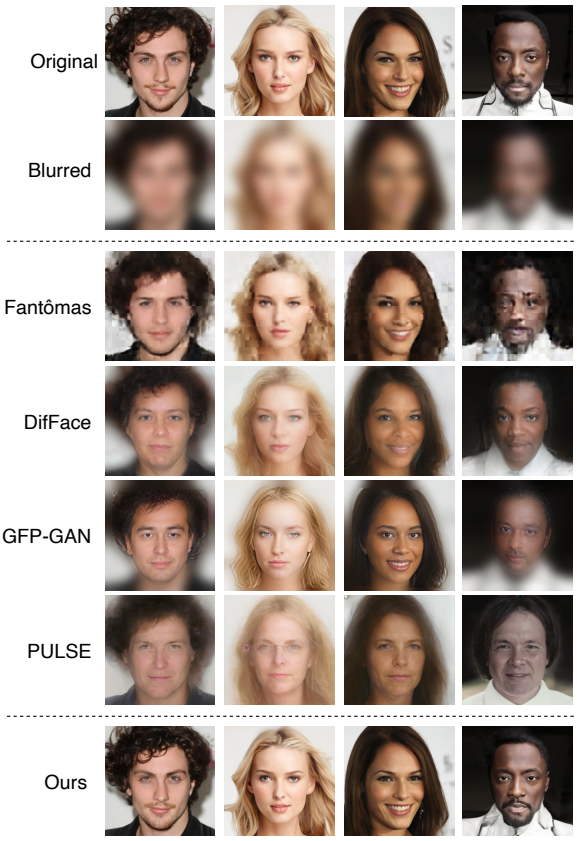

Figure 1: The original face has been blurred with Gaussian blur (kernel size $K = 81$) to hide the user identity. Our proposed method (`Revelio`) can restore the blurred face with a higher fidelity with respect to the original identity.

The key idea is to (1) use a conditional Diffusion model to approximate the inverse of the Gaussian blur function, and (2) use a large reference database and an identity retrieval model to augment the face restoration process. The intuition is two-fold. First, we utilize the memorization effect [19] in diffusion models. We assume the adversary can construct a large *reference database* D of face images. This is a realistic assumption given the aggressive image collection from social media services and search engines, which can also be done by other parties via web crawling. By training the diffusion model with such a reference database, we use the memorization effect to synthesize face details based on the matching identity. Importantly, even if the target identity is *not* in the reference database D, it is still possible for the model to perform face restoration with *similar-looking* identities in the reference database (validated in §5.5). Second, while it is impossible to construct a lossless inverse function for Gaussian blur, we modify the diffusion model to *approximate* the mapping from the blurred face to the original clear face. This is done by making the denoising process *conditioned* on the blurred input to preserve the original identity.

Based on these intuitions, `Revelio` takes a Gaussian blurred image x as input, and first performs a *preliminary restoration* to produce an image y_b using a base diffusion Model. Then it uses y_b to query the reference database to find a potentially matching identity. If the blurred image contains a known identity, the system retrieves reference images of this person and uses these images to *fine-tune* the base model to enhance the fidelity of face restoration. However, if the person is never seen (i.e., not in the reference database), `Revelio` can detect this is an out-of-distribution (OOD) identity, but still uses other "similar-looking" faces to recover the face details.

Evaluation. We evaluate `Revelio` using public face image datasets CelebA-HQ [20] and FFHQ [21]. We construct a reference database of 28,000 images and perform testing on images that *never appear* in the reference database. We test both light-blur (kernel size 37) and heavy-blur (kernel size 81) settings, and compare `Revelio` with existing face restoration solutions (GFP-GAN [14], DifFace [15], PULSE [17], and Fantômas [18]), and a Parrot Recognition method [22]. We have three main observations. First, `Revelio` outperforms existing methods with respect to the restored image quality, and more importantly, the *fidelity to the original identity*. The advantage of `Revelio` is more significant under the heavy-blur setting. Second, adversaries can use `Revelio` to successfully retrieve the identity of the blurred image when the identity is in the reference database. Under a heavy-blur setting, the identity retrieval accuracy achieves 95.9%. Importantly, even when the identity is *never seen* (OOD identity), `Revelio` can still use other similar-looking faces to restore the face (with minimal quality degradation). Third, `Revelio` exhibits robustness against images blurred with unknown Gaussian kernel sizes. Its built-in kernel size estimator can accurately infer the kernel size with a mean absolute error (MAE) lower than 1, and can tolerate kernel size mismatches within an offset of 6.

While defense is not the main focus of the paper, we have experimented with a few countermeasure ideas to disrupt the adversaries' models. We show that, while some of the countermeasures (e.g., image compression) can affect the vanilla attack, the impact can be largely mitigated by *adaptive attacks*. Further research is still needed to develop more secure and robust face anonymization methods. We have responsibly disclosed our findings to related parties including OpenCV, PyTorch, Adobe Photoshop, Apple (iOS SDK team) and FTC (Federal Trade Commission). Further ethics discussions are in §7.3.

Contributions. Our paper has three key contributions:

- **New method**: we developed an *identity-aware* deblurring method to restore Gaussian blurred face images. It combines a conditional Diffusion model with an identity retrieval method to restore the blurred face images while preserving identity fidelity.
- **New evaluation**: unlike prior works, we specifically target a *high-blurring* level to test the deblurring algorithm, which is more aligned with the face anonymization threat model. We show that our method outperforms existing solutions under such settings.

- **New tools**: we plan to responsibly share our code with other researchers to facilitate the development of defense methods. We have discussed our considerations regarding how to prevent potential abusive usage of the code

2. Background and Related Work

2.1. Gaussian Blur

Gaussian blur [5] is a widely used blurring method available in many popular photo-processing software such as Adobe Photoshop [8], Apple's Motion [9], Blur Photo Editor [10], PicsArt [11], and ASPOSE [12]. It is also supported by mobile SDK [23], [24]. Gaussian blur removes high-frequency details from the image by convolving the image with a Gaussian function. It superimposes a 2D Gaussian distribution over a group of pixels in the image and computes new RGB values by weighted averaging every pixel with its neighboring pixels. The weights are based on the Gaussian distribution. Below shows the Gaussian function in two dimensions:

$$G(i,j) = \frac{1}{2\pi\sigma^2} e^{-\frac{i^2+j^2}{2\sigma^2}} \quad (1)$$

This function creates the convolution *kernel* applied to every pixel in the original image. Here, i and j specify the location coordinates from the center pixel (0, 0). We use K to represent the convolution kernel size, which also defines the range of i and j. For example, if the kernel size $K = 3$, then i and j would range from -1 to 1 (inclusive). In this paper, we use a Gaussian blur with a square kernel. K needs to be a positive *odd integer* such that there is one center pixel in the kernel. Note that the Gaussian distribution is parameterized by σ (standard deviation), which also influences the strength of the blurring. A larger σ leads to a flatter/wider Gaussian distribution, leading to a stronger blurring effect. In a typical Gaussian blur implementation, σ needs to be scaled in proportion to the kernel size K. Otherwise, if we have a large kernel radius but a small σ, the far-away pixels would have little impact despite the large kernel size. As a result, real-world Gaussian blur implementations often use a fixed function to compute σ based on the user-specified kernel size. For example, both PyTorch [25] and OpenCV [26] implemented Gaussian blur using this dependency function: $\sigma = 0.3 * ((K-1) * 0.5 - 1) + 0.8$.

2.2. Blind Image Restoration

The machine learning community has worked on *blind image restoration*, which is related to but is different from our problem. Blind image restoration aims to recover high-quality images from degraded images without prior knowledge of the degradation. There are three main categories of methods. First, researchers have used Generative Adversarial Networks (GANs) for image restoration such as GFP-GAN [14], GPEN [27] and PSFR-GAN [28]. Second, researchers also use diffusion models for blind image restoration [16], [29], [30], [31], [32], some of which can be used on face images (e.g., DifFace [15]). However, they often focus on colorization, inpainting, uncropping, and JPEG restoration, but not Gaussian blur restoration for human faces. Third, super-resolution models [17], [33], [34] transform low-resolution images into high-resolution ones (e.g., PULSE [17]).

There are two key differences between the existing work and this paper. First, *application scenarios*. Most of these solutions are designed to handle unintentional image blur (e.g., caused by camera shakes), and their expected degradation strength is not very high. In contrast, we focus on *intentional* image blur for privacy protection. The expected blurring level needs to be high to hide facial identity. Second, *identity-awareness*. Most solutions are not designed to be identity-aware. The restored face details may not be those of the same person (Figure 1).

2.3. Image Restoration vs. Privacy Protection

Prior works from the security community have looked into how to restore images that have been blurred or pixelated for privacy/anonymization purposes. For example, Cavedon et al. [13] show that *pixelated* videos can be recovered using a Maximum A Posteriori (MAP) method. However, their inverse function is specifically designed for pixelization rather than Gaussian blur. They also require *multiple frames* in a video stream for the restoration. Hill et al. [7] introduce an attack method to recover the original text in redacted documents (from pixelization and blurring), but it does not handle human face images.

More recently, researchers have used machine learning methods to restore face images under anonymization. For example, Todt et al. [18] evaluated 15 image anonymization techniques (including Gaussian blur) and introduced Fantômas, an auto-encoder-based image restoration method. Unfortunately, this work did not perform face restoration under a *high degradation setting*. For Gaussian blur, they only used a small kernel size ($K = 29$). In §5, we show that Fantômas cannot effectively handle heavily blurred face images ($K = 81$) even after re-training.

Other related works proposed "Parrot Recognition" to directly re-identify anonymized images *without* performing image restoration [22], [35]. The idea is to train the classifier on blurred images and then perform classification on blurred images to recognize their identity. While the classification accuracy is higher than random guessing, it is still challenging for an accurate reidentification under a heavy-blur setting when there are a large number of candidate identities (e.g., over 1000). The accuracy is around 65% as shown in a recent work [22]. Considering that parrot recognition does not restore blurred faces, it only addresses part of our threat model (see §3 for details).

2.4. Facial Recognition and Attacks

There is a large body of related work on facial recognition systems [36], [37], [38], [39], [40], [41], [42], [43], and adversarial attacks against them [44], [45], [46], [47].

In addition, researchers have also re-purposed some of the attack methods as privacy protection tools [48], [49], [50], [51]. This line of work is different from ours because they operate on *clear* face images rather than Gaussian blurred images. Also, these privacy protection mechanisms have very different goals. They seek to fool facial recognition models, but the human faces in the images should still be visible and recognizable to human eyes.

3. Threat Model

We focus on photos that contain identifiable human faces. To protect user privacy, Gaussian blur is used to blur the *whole face* before the photo is posted on the Internet.

We assume the attacker has access to the blurred version of the photo (denoted as x), *but does not have access to* the original photo before blurring (denoted as y). The attacker's goal is to (1) restore the blurred photo such that the recovered face looks similar to the original face (i.e., face restoration), and (2) reveal the identity of the blurred face by matching it to a set of known identities (i.e., deanonymization). We believe that *deanonymization* and *face restoration* are two different levels of privacy violations, and achieving both presents a stronger attack.

In this case, we assume the attacker maintains a large *reference database* of human face images (denoted as D). The availability of such a dataset is a realistic assumption given the aggressive data collection of user images by social media services, search engines, AI companies, and even individuals via web crawling. The attacker only focuses on the face area—they can crop the blurred face region (e.g., in a square) before running the deblurring attack.

In our study, we consider two possible scenarios. First, in-distribution attack (or closed-world attack): the attacker's reference database D contains this victim's *other photos* (not y). The attacker thus aims to restore the face and link the known identity to the blurred image, achieving both deanonymization and face restoration. Second, out-of-distribution (OOD) attack (or open-world attack): the attacker's reference database D never indexes any photos of this victim. In this case, the attacker should be able to tell that the victim is not in the reference database. However, the attacker can still achieve face restoration, i.e., recovering the blurred face to look similar to the victim's original face (using other face images in D). While OOD attack cannot achieve *immediate* deanonymization, it still has practical implications through *face restoration*. For example, by publishing the "deblurred" photo, the person in the photo could be potentially re-identified in an ad-hoc manner by the viewers who know the person in real life (e.g., friends/families/supervisors). Also, if the target is a wanted criminal, law enforcement could use the restored photo (as a police sketch) for their investigation.

4. Methodology

We develop a system called `Revelio` to restore blurred face images and recover their original identities. The system design of `Revelio` is shown in Figure 2.

4.1. Intuitions and Design Overview

Challenges. First, Gaussian blur introduces significant "information loss" to an image as introduced in §2.1. Existing works such as DDRM [52] and SNIPS [53] treat the general image restoration problem as a *linear inverse* problem; however, the degradation matrix of Gaussian blur is ill-conditioned for this formulation due to its irreversible information loss [5]. ML-based restoration methods [14], [15] are not designed for intentionally applied heavy blurring for privacy reasons (§2.2).

Design Intuitions. We design `Revelio` based on two key intuitions. First, we utilize the memorization effect of generative models. Recent research [19], [54] shows that generated images from diffusion models are partially replicating the training data. In our scenario, this is not necessarily a weakness but an opportunity to construct attacks. Even though the model is *not* trained on the clear version of the input image, as long as the model has seen this person's other images, the memorization of training data could lead the model to reproduce this person's facial features. Second, while it is impossible to construct a lossless inverse function for Gaussian blur, we can use a diffusion model to *approximate* the inverse function. The information loss can be *partially* recovered by learning the features of general human faces, especially faces that look similar to the target.

4.2. Face Restoration

We start by constructing the Base Model M_B for face restoration (step ❷ in Figure 2). Given a blurred image x, M_B aims to restore it to a clear image that looks similar to the original image y.

Training: Overview. We train M_B using the *reference database* D owned by the adversary. Using D, we first construct the training dataset by applying Gaussian blur on the images to produce "ground-truth" pairs of blurred images x and their originals y as tuples: $\{(x, y)\}$. Using this dataset, the model learns the inverse mapping from the blurred image x to the corresponding clear image y. The idea is to approximate the conditional distribution $p(y|x)$ (which represents the inverse of Gaussian blur). Instead of using the traditional Denoising Diffusion Probabilistic Model (DDPM) [55] (which only captures the distribution $p(y)$ of generic face images), we use conditional diffusion [30] such that the generated faces are faithful to the original face's identity (i.e., conditioned by the input x).

Diffusion Model's Forward Process. A diffusion model has a forward process and a reverse (denoising) process. The forward process gradually adds Gaussian noise (not Gaussian blur) to "destroy" the clear image until the image turns into pure Gaussian noise. Then, we learn to recover the original image by modeling the reverse process. For our forward process, we start by drawing an image pair (x, y)

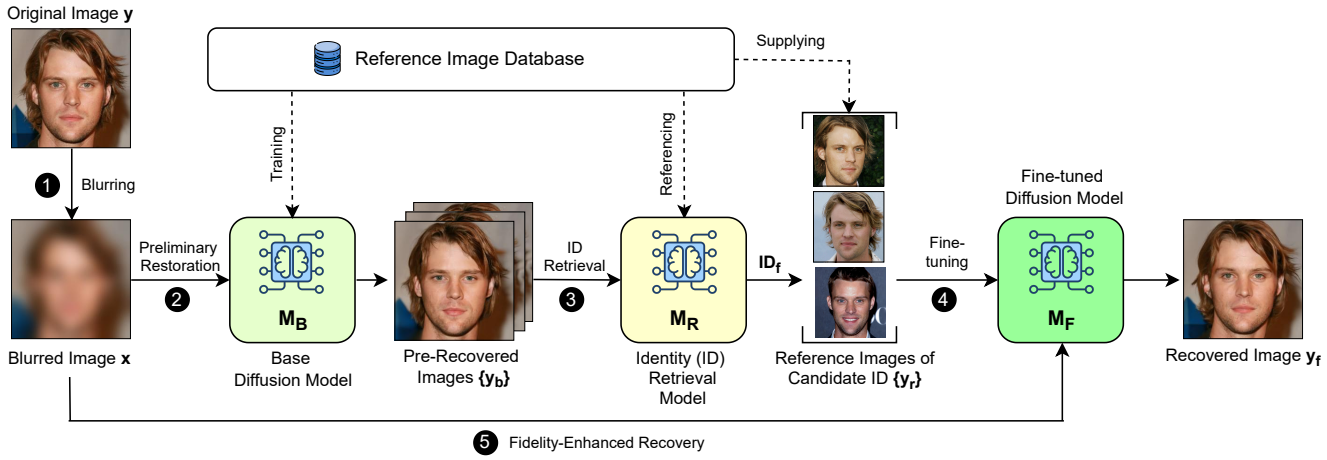

Figure 2: System workflow. The original clear image y is Gaussian blurred to generate image x (❶). This blurred image x is the input to `Revelio`. To recover the original face, we first use a Base Model M_B to perform preliminary restoration (❷). Then in step ❸, we use pre-recovered image y_b for identity retrieval. In step ❹, we use the matched identity to find more reference images to fine-tune the model. Finally, using the fine-tuned model (M_F), we can further improve the fidelity of the restored image (❺).

from the training dataset. We set $y_0 = y$, and then a forward Markov chain gradually adds Gaussian noise into y_0 with a total of T steps:

$$q(y_t|y_{t-1}) = \mathcal{N}(y_t|\sqrt{\alpha_t}y_{t-1}, (1-\alpha_t)\mathbf{I}) \tag{2}$$

Here t represents a time step from 1 to T. $\mathcal{N}$ and $\mathbf{I}$ denote a Gaussian distribution and the identity matrix, respectively. α_t is a hyper-parameter between 0 and 1 which defines the variance of the Gaussian noise added in each step. We can rewrite the forward process into a closed form. Given the initial clear image y_0 and y_t at any step, we can represent the distribution y_{t-1} as the following:

$$q(y_{t-1}|y_0, y_t) = \mathcal{N}(y_{t-1}|\mu, \sigma^2\mathbf{I}) \tag{3}$$

where $\mu = \frac{\sqrt{\gamma_{t-1}}(1-\alpha_t)}{1-\gamma_t}y_0 + \frac{\sqrt{\alpha_t}(1-\gamma_{t-1})}{1-\gamma_t}y_t, \sigma^2 = \frac{(1-\gamma_{t-1})(1-\alpha_t)}{1-\gamma_t}$, and $\gamma_t = \Pi_{s=1}^{t}\alpha_s$. Note that γ_t can represent the strength of the current noise level, which will be used in the denoising process later. μ and σ here are parameters of Gaussian noise (not Gaussian blur).

Conditional Denoising Process. For the de-noising process, we learn to recover the clear image y by optimizing a neural network model f_θ that is *conditioned* on the blurred image x. This model f_θ takes as inputs the blurred image x, a noisy image y_t at any step, and its current noise level γ_t, and predicts the noise vector ϵ. The noise vector ϵ will be used to denoise y_t to restore the previous version y_{t-1}. Here we use the same objective function for training the network f_θ used by [30]:

$$\mathbb{E}_{(x,y)}\mathbb{E}_{\epsilon,\gamma}\left\|f_\theta(x, y_t, \gamma_t) - \epsilon\right\|_2^2 \tag{4}$$

Optimizing Eqn. 4 is the key to face restoration. Here we use a U-Net architecture for f_θ, *concatenating the blurred image x with y_t as the input*. In this way, the blurred image x serves as a prior that contains information about what the clear image should look like. Without x, the model would denoise y_t blindly, leading to the loss of *identity* information (i.e., only generating generic human faces). By conditioning the denoising process on the blurred image x, the model learns to recover the clear image y while using the blurred x as guidance, preserving the identity of x. This denoising process learns to remove both the Gaussian noise and the Gaussian blur effect step by step. We also use the attention mechanism on the U-Net architecture. The U-Net architecture captures image features and expands them through multiple layers. At the bottleneck layer, the network encodes localized facial features (e.g., the eyes and the mouth). We apply an attention mechanism at this layer, encouraging the model to restore fine-grained face details.

Inference. After the above training process, we obtain a f_θ that can approximate y_{t-1} given x, y_t, and γ_t. Specifically, Eqn. 3 now can be rewritten as the following to calculate y_{t-1}:

$$y_{t-1} \leftarrow \frac{1}{\sqrt{\alpha_t}}\left(y_t - \frac{1-\alpha_t}{\sqrt{1-\gamma_t}}f_\theta(x, y_t, \gamma_t)\right) + \sqrt{1-\alpha_t}\epsilon_t \tag{5}$$

Thus, during the inference time, given an input blurred image x and a Gaussian noise vector y_T, we can restore and refine the image iteratively to recover facial details while preserving the identity of x. The model is trained on a large-scale face image dataset (D), and we expect the model to memorize high-fidelity details related to given identities. During the inference time, we expect the model to restore the facial details leveraging the memorization effect (e.g., using other images of x's identity in the database, or faces of other similar-looking identities).

Practical Consideration: Kernel Size Estimation. Note

that the above model M_B is trained under a specific kernel size for Gaussian blur. The kernel size K is set when we construct the training data for M_B. In practice, adversaries may not know the kernel size used by the input image x. In this case, adversaries can either (1) try different models (trained with different kernel sizes) and assess the recovered image quality, or (2) proactively estimate the kernel size based on input image x. We believe the second option is more cost-efficient. Below, we develop a kernel size estimator for this purpose.

Given an input blurred image x, the goal of the model is to infer the kernel size K. To train the model, we first construct a *synthetic training dataset* with "ground truth". This is done by randomly sampling face images and applying Gaussian blur on each image using a series of kernel sizes. Then we train a kernel size estimator using a regression model. Here, we use a pre-trained model EfficientNetV2 Large [56] as the pre-trained model to improve the model performance (pre-trained on ImageNet [57]). Then we replace the classification layer with a single output layer for regression to estimate the kernel size K. We obtain the model size K by rounding the regression estimation to the nearest odd integer value. Then the K will be used to select the proper model M_B for the face restoration of x.

4.3. Identity Retrieval

As shown in Figure 2 (❸), after using M_B to perform preliminary face restoration, we expect the recovered face image y_b to be somewhat similar to the original image but is not yet of high quality and fidelity. Our hypothesis is that image y_b is *good enough* to recover the *identity* of the target person if this person is in the reference database.

Face Embedding. To perform identity retrieval, the adversary first needs to map the face images in the reference database D to an embedding space. In this embedding space, face images of the same person (identity) should be clustered together while face images from different identities should be mapped further away. To do so, we utilize an open-source face recognition system [58] to perform face image embedding. We select this model for its high facial recognition performance as it achieves an accuracy of 99.38% on the LFW ("Labeled Face Images in the Wild") benchmark [59].

Identity Retrieval Function. A naive method is to compute the *average* distance between the input image y_b and other images of existing identities in the embedding space to find the nearest neighbors for identity detection. However, this naive idea is easily affected by the *stochastic process* of face generation of the diffusion model M_B. When we run M_B multiple times on the same input, the generated images have significant variance, especially when the blurring level is high. For certain rounds, the generated images are highly similar to the target person, while for other rounds the generated images look very different. The intuitive explanation is that the *memorization effect*, when triggered, can produce face details of the original person. However, the memorization effect is not always triggered the same way (or triggered at all) each round.

To this end, we determined that taking an *average distance* is not the best option. Instead, we run M_B on the same input multiple times (i.e., n rounds) to generate a set of $\{y_b\}$, and then rely on the *shortest distance* to determine the identity. The intuition is that within the multiple rounds of face generation, one or a few rounds will trigger a strong memorization effect to generate face images close to the true identity. We rely on these rounds (with the shortest distance) to maximize the success of identity retrieval. We take $n = 50$ as our default settings. For each pre-recovered image in the set $\{y_b\}$, we calculate the distance between the restored image and its nearest neighbor in the reference database. We then take the identity (ID) of the nearest neighbor as the winning identity for this round. Finally, the most frequently appearing winning identity among the n rounds would be selected as the final identity for input x, which is denoted as ID_f.

Detecting Out-of-Distribution (OOD) Identities. The above identity retrieval process will make a mistake when the identity of the blurred image x is never indexed in the reference database. In other words, the person in x has no other images in the reference database and is never seen by the diffusion model. We call such identities as out-of-distribution (OOD) identities. To detect blurred images of OOD identities, we use a simple shortest-distance threshold. More specifically, out of the n rounds of restoration, we identify the lowest distance between any of the pre-recovered images in $\{y_b\}$ and their nearest neighbors in the reference database. We denote this lowest distance as ld_x. If ld_x is higher than a threshold d, we will determine that x has an OOD identity. We will explore how the threshold affects OOD detection accuracy later in §5.5.

4.4. Fidelity Enhancement

Once the identity has been recovered, we retrieve related images from the *reference database* to further improve the fidelity of image restoration. This step corresponds to steps ❹ and ❺. The pre-recovered image y_b may still carry random facial features from the stochastic diffusion process. To further improve fidelity, we fine-tune the base model. First, based on the retrieved identity ID_f, we obtain a set of reference images of this identity from the reference database (denoted as $\{y_r\}$). These images are of the same person but do not contain the target image y. Second, using these images as the fine-tuning dataset, we adapt the attention block fine-tuning technique [60] to fine-tune M_B. We freeze all residual blocks in the U-Net backbone architecture in the base model except the attention blocks, which helps to preserve the identity information during fine-tuning. Finally, the fine-tuned model M_F can generate image y_f, which is expected to be of higher fidelity to the ground-truth image.

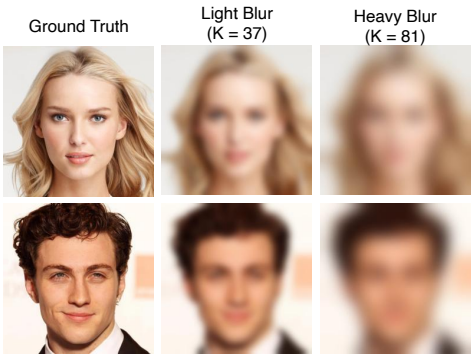

Figure 3: We use two different levels of Gaussian blur. The *light* blur setting has a kernel size $K = 37$. The *heavy* blur setting has a kernel size $K = 81$.

5. Evaluation

In this section, we evaluate the effectiveness of `Revelio` and compare it with existing solutions.

5.1. Dataset and Setup

Gaussian Blur Setup. We use PyTorch's implementation of Gaussian blur as the target for proof-of-concept. The same implementation is also used by OpenCV. We use kernel size K to control blur levels (the standard deviation σ of the Gaussian distribution is dependent on K). We use two different blur levels including light blur and heavy blur (see Figure 3 for examples). Given the image size of 256×256 images, we apply Gaussian blur with kernel size $K = 37$ for "light blur" and use kernel size $K = 81$ for "heavy blur". $K = 37$ roughly matches the degradation level of existing works. As discussed in §2.2, existing works used Gaussian blur as a generic way to degrade image quality (rather than for privacy protection), and thus did not use high blurring levels. As shown in Figure 3, under light blur, the blurred image does not fully hide the person's facial details. In contrast, under heavy blur, the images lose significantly more details, making it harder to re-identify the person. Our experiment will primarily focus on the high-blur setting given it is more challenging for attackers. In our experiments, if not otherwise stated, `Revelio` uses the matching kernel size to train the corresponding model. In §5.4, we will evaluate the attacker's ability to predict the kernel size and its tolerance of mismatched kernel sizes.

Datasets. Our experiment mainly uses the CelebA-HQ dataset [20], which includes 30,000 images, and 6,217 labeled identities. Another dataset FFHQ [21] is used for OOD evaluation (§5.5). We selected these datasets because they have been widely used by existing research [14], [15], [27], [28], [29], [52], making it easy for result reproduction. Both datasets contain high-quality face images. Importantly, CelebA-HQ includes *identity labels* (rare among public datasets). Each person (identity) has multiple images (5 images per identity on average). We did not use datasets such as MegaFace [61], [62] and LFW [59], either because the dataset was decommissioned or the image quality is low.

We randomly split the images in CelebA-HQ into the training set (28,000 images) and the testing set (2,000 images). The training set contains 6,084 unique identities, and the testing set contains 1,521 unique identities (with 1,388 identities overlapping between the training and testing sets). Despite the overlap of identities, a specific image can only appear in either the training set or the testing set but not both. We train our Base Model M_B with 28,000 training images for 1,000 epochs. Due to computing resource limitations, the dimensions of the images are set to 256×256 during training and testing. The *training set* (28,000 images) also serves as our reference database for identity retrieval.

Experiment Setup. Our experiment covers both closed-world (in-distribution attack) and open-world (OOD attack) settings. The main experiment uses the closed-world setting to assess the design choices in `Revelio`. Under this setting, the identities of the blurred images are *included* in the reference database. For open-world settings, we test identities that are *not included* in the reference database.

For the closed-world setting, we randomly select 50 identities and a total of 97 images from the *testing set* to apply Gaussian blur. These identities have other images in the reference database (i.e., the training set) but these 97 testing images never appear in the training set. For the open-world setting, we manually construct a set of out-of-distribution (OOD) identities and ensure (1) the testing images never appear in the training set; and (2) these identities have no other images in the training set either. In other words, these identities are never seen by the model. We obtain 97 of such images from 91 OOD identities (see §5.5 for details).

5.2. Evaluation Metrics and Baseline Methods

Evaluation Metrics. We use different metrics to evaluate the *identity retrieval* and *face restoration*. For identity retrieval (model M_R), we evaluate its performance using an *Identity Retrieval Accuracy* (IRA) which measures the ratio of blurred images for which the model correctly retrieves their identity out of all the testing images.

For the face restoration (models M_B and M_F), we use *standard* metrics to assess the fidelity of the restored images and the overall image quality. Fidelity evaluation measures how similar the restored image is in comparison with the original ground-truth image. First, we use PSNR [63] to make a similarity comparison at the pixel level for the two images. Second, we use LPIPS [64] and SSIM [65] metrics that approximate human perception for image similarity comparison. Third, we measure how well the restored image preserves the facial *identity* of the original image using ID Distance (IDD) [66]. Unlike the other metrics (that consider features of the *entire* images), IDD focuses more on facial features related to a person's identity (e.g., eyes and nose). IDD calculates the angular distance between the feature vectors of the restored face image and the ground-truth image (features extracted by a pre-trained ArcFace [36])

Kernel Size	K = 37	K = 81
Blurred	38.1%	0.0%
Parrot Recognition	60.8%	51.5%
Fantômas	35.0%	8.0%
GFP-GAN	26.8%	1.0%
PULSE	0.0%	0.0%
DifFace	37.1%	1.0%
Ours	**100%**	**95.9%**

TABLE 1: Identity retrieval accuracy. We test both the light-blur ($K = 37$) and heavy-blur ($K = 81$) settings.

Finally, to assess the overall quality of the restored images, we use FID [67] which measures the distribution similarity between the output datasets (of the restored images) and the ground-truth datasets. FID measures how well the restored faces resemble general human faces.

Comparison Baselines. We compare `Revelio` with existing restoration methods including Fantômas [18], GFP-GAN [14], DifFace [15], and PULSE [17]. Fantômas, GFP-GAN, DifFace use auto-encoder, GAN, and diffusion models, respectively, for face restoration. PULSE is a super-resolution model to restore face images. We chose these models because they are able to handle Gaussian blur, and their authors have made the code available for sharing. Fantômas was originally trained with kernel size K=29 and thus we retrained it with K=37 and 81. We also included a Parrot Recognition method [22] as an identity retrieval baseline. As described in §2.3, Parrot Recognition directly trains a reidentification classifier on *blurred* images without performing face restoration.

5.3. Basic Evaluation Results

We first run basic experiments to evaluate identity retrieval and face restoration under the closed-world setting (i.e., the target identity is included in the reference database).

Identity Retrieval Accuracy. We start by evaluating the identity retrieval model (Table 1). Before testing any face restoration methods, we first establish a baseline by directly running *our* identity retrieval method on the *blurred images* x without restoration. The result is shown in the "blurred" row in Table 1. The result confirms that "light blur" (with K=37) is not enough to fully hide the person's identity as the model can still recognize 38.1% of the *blurred faces*. However, under "heavy blur" (with K=81), none of these blurred faces are recognizable (0% accuracy). Table 1 also shows the performance of Parrot Recognition [22] where the classifier (ArcFace) has been retrained on blurred images (with matching kernels). While parrot recognition has an improved accuracy of 60.8% and 51.5% under the heavy and light blur settings, respectively, the accuracies are still not considered high.

Next, we examine identity retrieval accuracy on *restored* faces. For `Revelio`, we run the identity retrieval model (M_R) on the pre-recovered images (from M_B) to identify the person in the image. We test both the heavy-blur (K=81) and light-blur (K=37) settings. As shown in Table 1, our system achieves much better performance compared with existing methods. Under light-blur, our system can 100% recover the facial identity using the pre-recovered images from M_B. Under the heavy-blur setting, our system still achieves a 95.9% identity retrieval accuracy. Among the other baseline methods, faces restored by Fantômas, DiFace, and GFP-GAN preserve some identity information under the light-blur setting. However, under the heavy-blur setting, such identity information is no longer recovered, which leads to a low identity retrieval accuracy (lower than 8.0%). This is confirmed by the examples in Figures 1 and 6.

The few errors from our system are caused by the inherent challenge of distinguishing identities with similar facial features and makeup. Recall that our reference database contains 28,000 images and 6,084 identities, which makes face-matching difficult. We present case studies in Appendix A. As shown in Figure 11, the mismatched identity looks very similar to the restored face (as well as the original face).

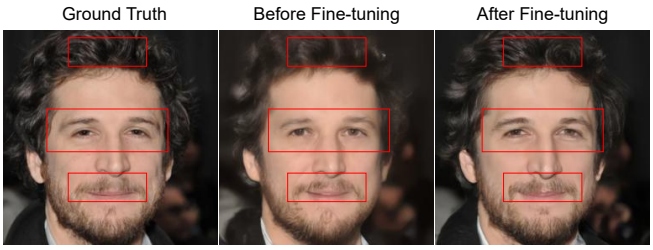

Figure 4: Restored faces *before* fine-tuning and *after* fine-tuning. We highlight the area where the fine-tuning has made a major improvement in image fidelity and quality.

Model	PSNR↑	SSIM↑	LPIPS↓	IDD↓	FID↓
M_B	24.75	0.68	0.24	0.71	41.84
M_F	**24.94**	**0.69**	**0.23**	**0.64**	**39.97**

TABLE 2: The Base Model M_B vs. the Fine-tuned Model M_F under the heavy-blur setting ($K = 81$). ↑ means a higher value is better. ↓ means a lower value is better.

Impact of Fine-Tuning. Next, we use experiments to demonstrate the impact of the Fidelity Enhancement module (i.e., fine-tuning) for `Revelio`. This is done by comparing the restored image quality and fidelity before and after running the fine-tuned model (M_F). For brevity, we only run this experiment for the heavy-blur setting ($K = 81$) because the image quality is already very good without fine-tuning under light blur (see Figure 5).

As shown in Table 2, the fine-tuned model M_F outperforms the base model M_B across all evaluation metrics. Among the five metrics, we observe noticeable improvements for IDD (which measures the fidelity of the face images with respect to the original *identities*). Through fine-tuning, we can effectively preserve the facial features of the original person and suppress the randomness introduced by the stochastic diffusion process.

Figure 4 shows a qualitative comparison between the

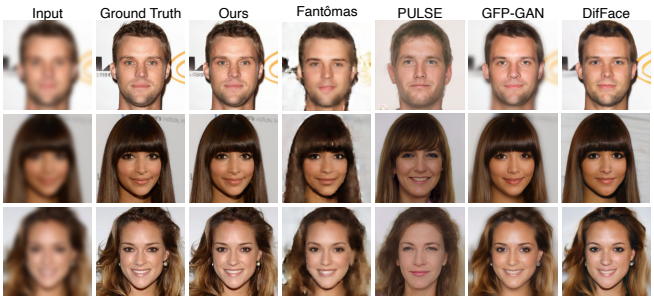

Figure 5: Face restoration under Light Blur (K = 37).

Methods	PSNR↑	SSIM↑	LPIPS↓	IDD↓	FID↓
Blurred	22.87	0.66	0.61	1.32	148.19
Fantômas	23.65	0.66	0.44	0.98	120.79
GFP-GAN	24.08	0.69	0.29	0.90	94.80
PULSE	18.78	0.54	0.43	1.37	104.72
DifFace	25.30	0.70	0.29	0.88	78.07
Ours	**28.04**	**0.78**	**0.17**	**0.37**	**27.16**
GT	∞	1	0	0	0.1

TABLE 3: Face restoration comparison under Light Blur (K = 37). "GT" denotes the ground-truth clear image.

output images of the two models. While the output before fine-tuning (from M_B) is already a high-quality face image, its fidelity to the original face is still slightly off. In this example, facial features related to the eyes and the mouth are still different from those of the original person. Also, the hair area still has some blurring effect. After fine-tuning, we can observe that the image (from M_F) has clear fidelity improvements in these facial features, making it more similar to the original person. Other subtle differences can be observed by zooming in on the example images.

Comparison with Existing Methods. Finally, we compare the performance of `Revelio` with existing methods. Under the light-blur setting ($K = 37$), we compare the results from our model (without fine-tuning) with the existing models. As shown in Table 3, our model (even without fine-tuning) outperforms all baseline methods across all evaluation metrics. The most noticeable improvements are shown by the LPIPS, FID, and IDD metrics. These metrics measure perceptual similarity, overall image quality, and identity-level fidelity. We present example images in Figure 5. Under the light-blur setting, existing methods demonstrate their ability to restore blurred images. The overall image quality is good even though the restored faces may not always look like the original person.

`Revelio`'s advantage is more visible under the *heavy-blur* setting ($K = 81$). We compare the results of existing models with our M_F model. As shown in Table 4, our model outperforms existing models on all evaluation metrics, with a much bigger gap than that under the light-blur setting. Figure 1 has already shown some example images, and we present additional examples in Figure 6 to cross-compare with the examples under light blur (Figure 5). On one hand, existing methods are not identity-aware. As a result,

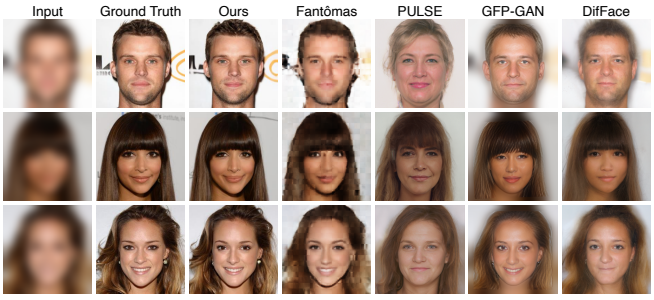

Figure 6: Face restoration under Heavy Blur (K = 81).

Methods	PSNR↑	SSIM↑	LPIPS↓	IDD↓	FID↓
Blurred	19.38	0.56	0.61	1.50	290.51
Fantômas	22.06	0.61	0.49	1.16	134.61
GFP-GAN	19.89	0.57	0.38	1.28	123.68
PULSE	17.25	0.50	0.46	1.42	116.78
DifFace	20.49	0.57	0.38	1.26	99.94
Ours	**24.94**	**0.69**	**0.23**	**0.64**	**39.97**
GT	∞	1	0	0	0.1

TABLE 4: Face restoration comparison under Heavy Blur (K = 81). "GT" denotes the ground-truth clear image.

they cannot faithfully restore the identity of the original person in the image. In contrast, our method can achieve identity-aware face restoration, by combining a conditional diffusion model with identity retrieval. On the other hand, existing methods are not designed to recover faces from heavy blurring. Their performance is worse under such settings. Overall, the result confirms that `Revelio` can handle severely blurred images that the state-of-the-art face restoration models are unable to (or not designed to) address.

5.4. Impact of Mismatched Kernel Size

So far, our experiments have assumed that the adversary knows the kernel size K of the Gaussian blur. In practice, the adversary will need to infer this information based on the blurred input. In §4.2, we described the methodology for kernel size estimation. Here, we evaluate its effectiveness.

Estimating Kernel Size Based on Blurred Images. To train the kernel size estimator model, we randomly sample 1,000 images from the CelebA-HQ's *training set*, and split the data randomly into training, validation, and testing sets with 800, 100, and 100 images, respectively. Then, we apply Gaussian blur with different kernel sizes on each image. We set kernel size K by enumerating all the odd numbers from 1 to 81 to generate blurred images. As mentioned in §2.1, the kernel size needs to be an odd number to align the center pixel. We train the regression model using the training set, adjust hyperparameters using the validation set, and test the model accuracy on the testing set.

The result shows that adversaries can *accurately* predict the kernel size based on the blurred image. The mean absolute error (MAE) is only 0.934, meaning that the model

can predict the kernel size K with an offset lower than 1. For example, if the ground-truth kernel size is 37, most predictions either fall on the correct kernel size of 37, or fall on the two neighboring kernel sizes (35 or 39). The result is further visualized in Figure 13 in Appendix C. This means adversaries can estimate the kernel size based on input images, and select the right model (trained with the matching kernel size) for image restoration.

Tolerance of Mismatched Kernel Sizes. Next, we further examine how much `Revelio` can tolerate mismatched kernel sizes. For this experiment, we select the heavy-blur setting (i.e., the more challenging setting) and train the Base Model M_B with Gaussian blur using a kernel size of 81. Then for the closed-world testing images, we apply Gaussian blur with varied kernel sizes ranging from 71 to 91. We show that the model can easily tolerate kernel size mismatches within the mean absolute error (MAE) of 6, under which the restored images are still of high quality and fidelity. Visual examples and quantitative metrics are presented in Figure 14 and Figure 15 in Appendix C. Given the above kernel size estimation model achieves an MAE below 1, we argue that the proposed method can sufficiently tolerate this level of mismatch.

In addition, we further test a different dependency function between K and σ for Gaussian blur. As mentioned in Section 2.1, most Gaussian blur implementations have a fixed dependency function between these two parameters, and thus attackers only need to predict K. If future implementations change the dependency function, does `Revelio` still work? In Appendix C, we have tested different dependency functions between σ and K and have an interesting observation. That is, regardless of the $K-\sigma$ dependency, as long as the blurring effect is similar to what `Revelio` is trained on, the system still works. This means adversaries can use the kernel estimator to blindly predict a kernel size K that has a similar blurring effect, and then select the corresponding M_B for restoration. For example, we use a different dependency function (K=27 and $\sigma = 6$) to blur the image. Our kernel estimator predicts $K = 39$ (meaning, the blurring effect is similar to $K = 39$ under the *old dependency function*). Then we choose M_B trained for the "light-blur" setting (with a close kernel size of 37) and it works well. Appendix C also includes additional transferability evaluation for a *non-squared kernel* for Gaussian blur and has a similar conclusion.

5.5. Open-world Setting Experiment

In this section, we evaluate the open-world setting where the input blurred image x has an OOD identity. In other words, the reference database D does not contain images of the person pictured in x. As discussed in §3, in this case, the attacker can still attempt to restore the victim's face. In practice, the deblurred photos can still lead to ad-hoc deanonymization, e.g., by viewers who know the victim in real life, such as families, friends, and colleagues.

We collected OOD identities and their images in two ways. (1) From the *testing set* of CelebA-HQ, we randomly sample images whose identities only appear in the testing set. In other words, these identities do not have any images in the training set (57 images of 51 identities). (2) We take another public face image dataset FFHQ [21] and randomly sample 40 identities (and their 40 images). We manually verified that these identities never appear in the reference database. In total, we have 97 images from 91 OOD identities. The OOD dataset is of the same size as the in-distribution set used by the closed-world experiment.

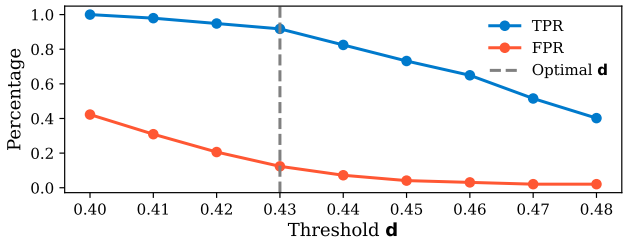

Figure 7: The true positive rate (TPR) and false positive rate (FPR) under different thresholds d. The gray line denotes $d = 0.43$ which yields an overall accuracy of 90%.

Setting	PSNR↑	SSIM↑	LPIPS↓	IDD↓	FID↓
OOD (Detected)	24.92	0.70	0.25	0.86	51.87
OOD (Missed)	26.12	0.74	0.23	0.78	56.58
In-Distribution	24.75	0.68	0.24	0.71	41.84

TABLE 5: Face restoration quality comparison between OOD identities (including correctly detected and missed OOD identities) and in-distribution identities ($K = 81$).

Our experiment has two goals. First, we test if we can *detect* OOD identities from in-distribution identities. Second and *more importantly*, even when an identity is OOD, we examine if `Revelio` can still *restore* the blurred face.

Detecting OOD Identities. As discussed in §4.3, we apply a threshold d on the lowest distance (ld_x) between the pre-recovered image and the reference images, to determine if the blurred image x has an OOD identity. Using the OOD and In-distribution datasets described above, we perform a classification of OOD images under the heavy-blur setting ($K = 81$) given it is more challenging for attackers.

Figure 7 shows the trade-off between the true positive rate (TPR) and the false positive rate (FPR) when the threshold d varies. When we set $d = 0.43$, we obtain a TPR of 0.92, an FPR of 0.12, and an overall detection accuracy of 90%. The result confirms we can detect OOD identities from the blurred images with a decent accuracy. We emphasize that detecting the OOD identities is only to provide *contexts* for the adversaries, that is, this person is likely not indexed in the reference database. The more important task is still to restore the blurred face, despite it being an OOD identity.

Face Restoration for OOD Identities. Given OOD identities are not indexed by the reference database, we examine the face restoration result from the Base Model (M_B). Table 5 reports the image quality and fidelity metrics for

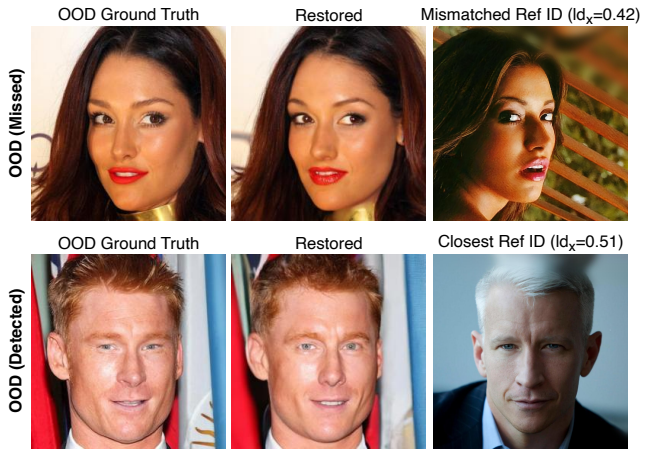

Figure 8: Face restoration examples for out-of-distribution (OOD) identities. For both examples, face restoration is successful as the restored face (middle) looks similar to the ground-truth image (left). The top row shows an OOD image that we failed to detect because the reference identity (right) looks too similar to the restored image (middle). The bottom row shows a correctly detected OOD image because the closest reference identity looks different enough.

OOD identities including those that are correctly detected by the detector (using threshold $d = 0.43$) and those that are missed. We also show the restoration results for "in-distribution" identities as a reference.

We find that the face restoration is highly successful. As shown in Table 5, the restored image quality and fidelity are comparable to those of the in-distribution identities. This observation applies to both correctly detected OOD identities and those that are missed.

Figure 8 also shows that the restored faces (middle column) look highly similar to the ground-truth images before blurring (left column). We present extra examples in Figure 12 in the Appendix B. This result shows `Revelio` is able to recover the blurred face even if the corresponding person is never indexed in the reference database. A possible explanation is that `Revelio` in part learns to approximate the inverse of Gaussian blur. In addition, it may have used other "similar-looking" identities to synthesize the facial features for the target image. Figure 8 shows the closest reference images (right column) in the reference database for both examples. Although the OOD identity does not exist in the reference database, there exist "similar-looking" faces in the reference database that can help with face restoration. On one hand, such similar-looking faces are the main reason for OOD detection errors (the top-row example in Figure 8). On the other hand, this supports our intuition as to why face restoration is still possible on *previously unseen* identities.

6. Defense

Finally, we briefly explore possible countermeasures against the deblurring algorithm and explore adaptive attacks. We want to emphasize that robust defense is not the main focus of this paper. We want to explore a few ideas to inspire future directions.

Defense	PSNR↑	SSIM↑	LPIPS↓	FID↓	IRA↑
Rotation	27.39	0.76	0.23	51.69	87.62%
Gaussian Noise	**11.30**	**0.07**	**0.74**	**425.54**	**0%**
JPEG Compress.	17.15	0.25	0.63	376.37	**0%**
Box Blur	14.62	0.23	0.59	420.76	**0%**
No Defense	28.04	0.78	0.17	27.16	100%

TABLE 6: Comparison of different defense approaches (against M_B, $K = 37$). To be consistent with other tables, ↑ means that a higher value is better for *attackers* (i.e., worse for defenders).

Attack	PSNR↑	SSIM↑	LPIPS↓	FID↓	IRA↑
Adaptive Attack	25.71	0.73	0.22	25.71	88.66%

TABLE 7: Impact of adaptive attack against JPEG Compression Defense ($K = 37$). ↑ means that a higher value is better for *attackers*.

The goal of the experiment is to examine how sensitive the deblurring algorithm is to different post-processing steps and mismatched blurring algorithms. More specifically, after Gaussian blur is applied to the image, users can additionally process the blurred image by performing image rotation, adding Gaussian noises, and performing JPEG compression. The user may also use a different blurring algorithm as well. This will create a mismatch between the blurred image and the trained model of the attacker. Our experiment seeks to reveal what types of countermeasures have a major impact on image restoration quality. We run the defense experiments under the light-blur setting ($K = 37$), which is a more challenging setting for *defenders*.

Disrupting Face Restoration Model. To disrupt the face restoration model, we apply image rotation, Gaussian noise, and JPEG compression on the blurred image x. We also test Box Blur (the OpenCV implementation) [26], a different blurring algorithm, to examine its disruption effect. Appendix D includes further details for the specifications of these defense methods. The restoration result is shown in Table 6. We also present visual examples in Figure 9. We show that the face restoration is robust against image rotation, but can be majorly disrupted by Gaussian noises, JPEG compression, and box blur. The restored image quality is significantly lower after these post-processing steps (especially for Gaussian noise). In Table 6 (the last column), we also report the identity retrieval accuracy (IRA) using the restored faces. For rotation, the IRA is still high (87.62%). The other defense methods can drop the IRA to 0%, confirming their effectiveness.

Adaptive Attacks. We want to emphasize that this experiment result *does not* mean the defense will be equally effective against adaptive attackers. To demonstrate this, we implement an *adaptive attack* against the JPEG Compression (one of the seemingly effective countermeasures). This is done by fine-tuning our Base Model M_B with images that

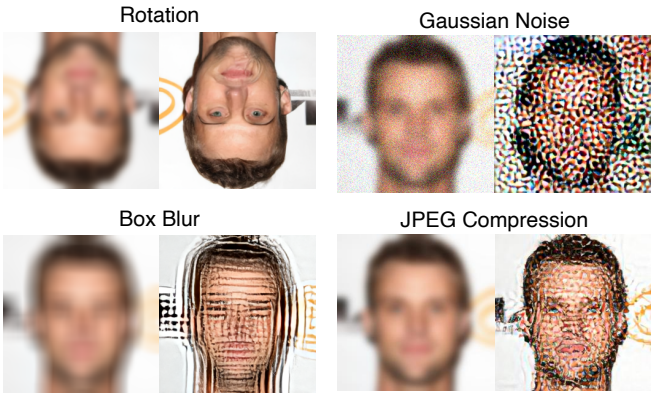

Figure 9: Defense against face restoration. We show that the face restoration method is robust against image rotation, but can be disrupted by other defense methods.

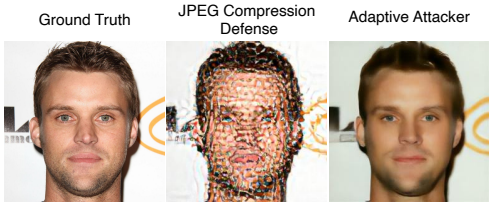

Figure 10: Adaptive attack against JPEG Compression defense. The result shows that the adaptive attack can effectively restore the blurred face.

are processed with Gaussian Blur and JPEG compression (for 100 epochs). The results of the adaptive attack are presented in Table 7, and example images are provided in Figure 10. With the *adaptive* attack, we observe that the restored face quality is much better. Importantly, the identity retrieval accuracy (IRA) improves from 0% back to 88.66%. The result confirms that the adaptive attack is successful.

As a countermeasure to the adaptive attack, defenders may further randomize the post-processing configurations (including the choice of post-processing methods and their parameters) as a "secret", and apply a different blurring process each time for each image. This may make a pre-trained deblurring model less effective on all blurred images. However, we argue that one can also choose *not to play this cat-and-mouse game*, for example, by covering the full face using a black/white square. In this way, zero information is left on the photo that is associated with the identity of the target person.

7. Discussion

7.1. Implications

Key Findings. Our paper provides concrete evidence that Gaussian blurred face images, even under a *high blurring* level, can be restored to their clear form, and can be used to re-identify the persons in the images. We also show a reasonable level of robustness of the deblurring process against unknown kernel sizes. Also, face restoration is possible, not only for *known* identities, but also for *previously unseen* identities that are not indexed in the reference database.

Privacy-Utility Tradeoff? Today, Gaussian blur is widely used by lay users and even journalists on sensitive photos in the real world [1], [2], [3], [4]. This may be due to the wide availability of Gaussian Blur tools, potentially misleading articles/recommendations on the Internet, and a lack of security awareness among users. When applying Gaussian blur, is there a meaningful trade-off between *privacy* and *utility*? From the users' perspective, there might be a trade-off since lightly blurred faces can still preserve information such as gender, hair color, and skin tone, making the photo appear more "authentic" [18]. However, the light blur makes face restoration and denonymization easier.

The next question is, would a higher blur level be safe to use? We can briefly reason the above question from the perspective of *conditional diffusion models*. Our model can deblur image x to preserve x's identity by conditioning the denoising process on x. As long as x still carries some information from the original image y, with sufficient training, it should be feasible to learn the mapping from x to y. A higher blurring level may make the training process more expensive, but it should not be impossible. To make Gaussian blur "safe", x should be completely independent of the clear image y. In this extreme case, for example, x can be a black image with all pixel values set to 0. In other words, we need to cover the *full face* with a black mask instead of blurring the face. We want to emphasize that covering the *partial face* (e.g., using eye masks) does not work, because the remaining face areas are still potentially re-identifiable [18]. Based on this reasoning, we believe Gaussian blur *should not* be used when privacy/anonymity is the primary concern.

7.2. Recommendations

Based on these findings, we make the following recommendations to users, software vendors, and policy makers.

Users. Users who need to anonymize faces in sensitive photos should not use the Gaussian blur, especially those implemented by popular photo process software (e.g., Photoshop). Such software can be easily studied by attackers to train *targeted* deblurring algorithms. This recommendation applies to both lay Internet users and professionals (e.g., journalists) who handle and publish photos of at-risk user populations (e.g., protesters). However, this does not mean other alternative blurring or degradation algorithms (e.g., pixelization, eyemasks, box blur) are secure. These alternative methods are not the focus of this paper, and thus we cannot speak to their security. However, related works have expressed similar concerns about their ability to hide the target information [7], [13].

Photo-processing Software Vendors. We recommend photo-processing software adding a *warning message* under

the Gaussian blur function (and other similar functions). This is to remind or inform users that Gaussian blur is not safe to be used to anonymize faces in sensitive photos. Such a warning message not only protects users from the threat of deanonymization but can also reduce the liability risks of software vendors. The warning should not affect the normal use of the blurring function for non-security/privacy scenarios.

Policy Makers. Face anonymization techniques that have known risks of deanonymization (including Gaussian blur) should be discouraged from being used in safety-critical and privacy-sensitive scenarios such as journalism and legal systems. Pushing new standards and policies will require close collaborations between the lawmakers and the technical community.

7.3. Ethics Consideration

We are mindful of the ethical implications of our research activities and the results. We believe our research's potential benefits (e.g., discovering vulnerabilities, raising user awareness, improving security practices) outweigh the potential risks.

Risk vs. Benefit Reasoning. Our work follows the common practice and the basic principles of *offensive security research* [68], [69]. Like many prior works [18], [44], [45], [46], [47], our goal is to reveal the security problem in existing solutions and improve the security practice before attackers independently discover and exploit the vulnerability at a large scale against unprepared, vulnerable targets. We argue that a "false sense of security" is worse than "known insecurity." In our case, without this type of research (including related prior works [7], [18], [35]), users may falsely believe that applying a Gaussian blur to their photos is sufficient to hide their identities. This false sense of security could lead to oversharing behaviors on the Internet, exposing more sensitive photos of users, and thus increasing the risk. A key benefit of our research is to provide concrete evidence on the security risk of applying Gaussian blur as a privacy protection mechanism and increase the awareness of related users (e.g., Internet users, journalists, activists). In addition, the result can potentially inform software vendors and policymakers to use or promote more robust privacy protection mechanisms. Our experiments require using datasets of human face images. We understand that human face data is a sensitive type of data. In our study, we limit ourselves to only using CelebA-HQ [20] and FFHQ [21], which are two publicly available benchmark datasets used by common machine learning research [14], [15], [16], [27], [28], [29], [30], [31], [32], [52], [70].

Responsible Disclosure. We reached out to related parties, including OpenCV, PyTorch, Adobe Photoshop, Apple (iOS SDK team), and FTC (Federal Trade Commission) to disclose our findings and share our recommendations. So far, we have received acknowledgements from Apple, PyTorch, and Adobe, with corresponding teams looking into the issues. We will document further details of our interactions with these vendors before the paper publication.

Code/Data Sharing. We will make our research artifacts (code, datasets) available for sharing with other researchers. However, we do not want malicious parties to use the code to cause harm. As safeguards. we will ask requesters to fill out a short form to explain how they plan to use the code and data. We will also verify the requester's identity and affiliation before sharing.

7.4. Limitations and Future Work

Our paper is limited in several aspects. Here, we discuss open questions and opportunities for future work.

Even Higher Blurring Levels. Our experiments use a high blurring level for Gaussian blur (K=81 for 256×256 images). This blurring level already renders a 0% accuracy for facial recognition (see Table 1). While we did not test a higher blur level, based on our reasoning analysis in §7.1, restoring face under a higher blurring level will require more expensive training but should not be impossible. We leave experimentation to future work.

Security-Aware Face Anonymization. Researchers have worked on face/image anonymization techniques with provable privacy guarantees [71], [72], [73]. Considering most of these systems are still research prototypes (i.e., not widely used in commercial products yet), we prioritize the analysis of Gaussian blur in this paper. Future research should further investigate the privacy guarantee of these anonymization methods under practical threat models, especially taking into consideration the emerging generative models and the availability of large image datasets with *clear* face images.

Dataset and Evaluation. Our experiment is limited by the datasets we use. First, the CelebA-HQ and FFHQ datasets contain high-resolution face images that are mostly front-facing. Future work should further investigate the feasibility of deblurring low-resolution face images or images with side faces. Further, future work can examine the impact of the reference database (e.g., in terms of its size and image quality) on the attack effectiveness. Second, we select CelebA-HQ because it contains identity labels on a large number of diverse face images. However, the number of images per identity is low (5 on average), and the distribution is highly skewed. This creates a challenge to the identity-retrieval model because there are not enough reference images for stable/reliable identity-matching. The identity retrieval process can be further improved with a dataset that contains more images per identity.

8. Conclusion

In this paper, we developed a system called `Revelio` and used a conditional diffusion model to restore Gaussian blurred face images. With extensive experiments, we showed that Gaussian blurred faces, even under a high blurring level, can be restored to their clear form and used to perform accurate re-identification. We showed that `Revelio` can handle

input images blurred with an unknown kernel size and the face restoration can be applied to both known identities and previously unseen identities. Based on our findings, we explored preliminary countermeasures and provided recommendations to users, software vendors, and policy makers.

References

[1] M. Lenthang and A. Mullen, "Video shows florida police sergeant grabbing fellow officer by her throat," https://www.nbcnews.com/news/us-news/florida-police-sergeant-accused-grabbing-officer-throat-rcna12236, 2022.

[2] C. Von Quednow and C. Bailey, "Officers find 'no real firearms' in home of wisconsin teen arrested on suspicions of planning a school shooting," https://www.cnn.com/2024/11/07/us/wisconsin-elementary-school-staff-stops-boy-with-suspicious-bags-entering-campus, 2024.

[3] R. Reilly, "Speaker mike johnson says he's blurring jan. 6 footage so rioters don't get charged," https://www.nbcnews.com/politics/congress/speaker-mike-johnson-says-blurring-jan-6-footage-rioters-dont-get-char-rcna128181, 2023.

[4] M. Aggeler, "Face of a dissident as images from protests circulate online, some fear that individuals will become targets," https://www.thecut.com/2020/06/face-of-a-dissident.html, 2020.

[5] R. A. Hummel, B. Kimia, and S. W. Zucker, "Deblurring gaussian blur," *Computer Vision, Graphics, and Image Processing*, vol. 38, no. 1, pp. 66–80, 1987.

[6] M. Bethany, A. Seong, S. H. Silva, N. Beebe, N. Vishwamitra, and P. Najafirad, "Towards targeted obfuscation of adversarial unsafe images using reconstruction and counterfactual super region attribution explainability," in *Proc. of USENIX Security*, 2023.

[7] S. Hill, Z. Zhou, L. Saul, and H. Shacham, "On the (in) effectiveness of mosaicing and blurring as tools for document redaction," in *Proc. of PETS*, 2016.

[8] A. G. Kenton Waltz, "Demystifying gaussian blur." https://www.adobe.com/creativecloud/photography/discover/gaussian-blur.html, 2024.

[9] Apple, "Motion user guide," https://help.apple.com/pdf/motion/en_US/motion-user-guide.pdf, 2025.

[10] Google-Play, "Blur photo editor: blur effect," https://play.google.com/store/apps/details?id=com.fookiemonsters.photo_blur_mosaic, 2025.

[11] D. Blake, "How to blur face in picsart," https://picsartone.com/how-to-blur-face-in-picsart/, 2025.

[12] ASPOSE, "Add gaussian blur filter to image," https://products.aspose.app/imaging/photo-filter/gaussian-blur, 2025.

[13] L. Cavedon, L. Foschini, and G. Vigna, "Getting the face behind the squares: Reconstructing pixelized video streams," in *Proc. of WOOT*, 2011.

[14] X. Wang, Y. Li, H. Zhang, and Y. Shan, "Towards real-world blind face restoration with generative facial prior," in *Proc. of CVPR*, 2021.

[15] Z. Yue and C. C. Loy, "Difface: Blind face restoration with diffused error contraction," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 46, no. 12, pp. 9991–10 004, 2024.

[16] B. Xia, Y. Zhang, S. Wang, Y. Wang, X. Wu, Y. Tian, W. Yang, and L. Van Gool, "Diffir: Efficient diffusion model for image restoration," in *Proc. of ICCV*, 2023.

[17] S. Menon, A. Damian, S. Hu, N. Ravi, and C. Rudin, "Pulse: Self-supervised photo upsampling via latent space exploration of generative models," in *Proc. of CVPR*, 2020.

[18] J. Todt, S. Hanisch, and T. Strufe, "Fantômas: Understanding face anonymization reversibility," in *Proc. of PETS*, 2024.

[19] G. Somepalli, V. Singla, M. Goldblum, J. Geiping, and T. Goldstein, "Understanding and mitigating copying in diffusion models," in *Proc. of NeurIPS*, 2023.

[20] T. Karras, T. Aila, S. Laine, and J. Lehtinen, "Progressive growing of GANs for improved quality, stability, and variation," in *Proc. of ICLR*, 2018.

[21] T. Karras, S. Laine, and T. Aila, " A Style-Based Generator Architecture for Generative Adversarial Networks ," *IEEE Transactions on Pattern Analysis & Machine Intelligence*, vol. 43, no. 12, pp. 4217–4228, 2021.

[22] S. Hanisch, J. Todt, J. Patino, N. Evans, and T. Strufe, "A false sense of privacy: Towards a reliable evaluation methodology for the anonymization of biometric data," in *Proc. of PETS*, 2024.

[23] Google, "Android sdk: Gaussianblur," https://developer.android.com/reference/androidx/media3/effect/GaussianBlur, 2025.

[24] Apple, "Developer documentation: Blur filters," https://developer.apple.com/documentation/coreimage/blur_filters, 2025.

[25] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer, "Automatic differentiation in pytorch," in *Proc. of NeurIPS Autodiff Workshop*, 2017.

[26] G. Bradski, "The OpenCV Library," *Dr. Dobb's Journal of Software Tools*, 2000.

[27] T. Yang, P. Ren, X. Xie, and L. Zhang, "Gan prior embedded network for blind face restoration in the wild," in *Proc. of the CVPR*, 2021.

[28] C. Chen, X. Li, L. Yang, X. Lin, L. Zhang, and K.-Y. K. Wong, "Progressive semantic-aware style transformation for blind face restoration," in *Proc. of CVPR*, 2021.

[29] X. Lin, J. He, Z. Chen, Z. Lyu, B. Dai, F. Yu, Y. Qiao, W. Ouyang, and C. Dong, "Diffbir: Toward blind image restoration with generative diffusion prior," in *Proc. of ECCV*, 2025.

[30] C. Saharia, W. Chan, H. Chang, C. Lee, J. Ho, T. Salimans, D. Fleet, and M. Norouzi, "Palette: Image-to-image diffusion models," in *Proc. of SIGGRAPH*, 2022.

[31] Y. Wang, J. Yu, and J. Zhang, "Zero-shot image restoration using denoising diffusion null-space model," in *Proc. of ICLR*, 2023.

[32] Y. Zhu, K. Zhang, J. Liang, J. Cao, B. Wen, R. Timofte, and L. Van Gool, "Denoising diffusion models for plug-and-play image restoration," in *Proc. of CVPR*, 2023.

[33] C. Saharia, J. Ho, W. Chan, T. Salimans, D. J. Fleet, and M. Norouzi, "Image super-resolution via iterative refinement," *IEEE Transactions on Pattern Analysis & Machine Intelligence*, vol. 45, no. 4, pp. 4713–4726, 2023.

[34] H. Li, Y. Yang, M. Chang, S. Chen, H. Feng, Z. Xu, Q. Li, and Y. Chen, "Srdiff: Single image super-resolution with diffusion probabilistic models," *Neurocomputing*, vol. 479, pp. 47–59, 2022.

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

[36] J. Deng, J. Guo, N. Xue, and S. Zafeiriou, "Arcface: Additive angular margin loss for deep face recognition," in *Proc. of CVPR*, 2019.

[37] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf, "Deepface: Closing the gap to human-level performance in face verification," in *Proc. of CVPR*, 2014.

[38] H. Wang, Y. Wang, Z. Zhou, X. Ji, D. Gong, J. Zhou, Z. Li, and W. Liu, "Cosface: Large margin cosine loss for deep face recognition," in *Proc. of CVPR*, 2018.

[39] W. Liu, Y. Wen, Z. Yu, M. Li, B. Raj, and L. Song, "Sphereface: Deep hypersphere embedding for face recognition," in *Proc. of CVPR*, 2017.

[40] F. Schroff, D. Kalenichenko, and J. Philbin, "Facenet: A unified embedding for face recognition and clustering," in *Proc. of CVPR*, 2015.

[41] Y. Sun, C. Cheng, Y. Zhang, C. Zhang, L. Zheng, Z. Wang, and Y. Wei, "Circle loss: A unified perspective of pair similarity optimization," in *Proc. of CVPR*, 2020.

[42] Q. Meng, S. Zhao, Z. Huang, and F. Zhou, "Magface: A universal representation for face recognition and quality assessment," in *Proc. of CVPR*, 2021.

[43] M. Kim, A. K. Jain, and X. Liu, "Adaface: Quality adaptive margin for face recognition," in *Proc. of CVPR*, 2022.

[44] Z. Wu, Y. Cheng, S. Zhang, X. Ji, and W. Xu, "Uniid: Spoofing face authentication system by universal identity," in *Proc. of NDSS*, 2024.

[45] G. Garofalo, V. Rimmer, D. Preuveneers, W. Joosen *et al.*, "Fishy faces: Crafting adversarial images to poison face authentication," in *Proc. of WOOT*, 2018.

[46] M. Sharif, S. Bhagavatula, L. Bauer, and M. K. Reiter, "Accessorize to a crime: Real and stealthy attacks on state-of-the-art face recognition," in *Proc. of CCS*, 2016.

[47] Y. Li, K. Xu, Q. Yan, Y. Li, and R. H. Deng, "Understanding osn-based facial disclosure against face authentication systems," in *Proc. of ASIACCS*, 2014.

[48] S. Shan, E. Wenger, J. Zhang, H. Li, H. Zheng, and B. Y. Zhao, "Fawkes: Protecting privacy against unauthorized deep learning models," in *Proc. of USENIX Security*, 2020.

[49] S. Jin, H. Wang, Z. Wang, F. Xiao, J. Hu, Y. He, W. Zhang, Z. Ba, W. Fang, S. Yuan *et al.*, "{FaceObfuscator}: Defending deep learning-based privacy attacks with gradient descent-resistant features in face recognition," in *Proc. of USENIX Security*, 2024.

[50] M. Chen, Z. Zhang, T. Wang, M. Backes, and Y. Zhang, "FACE-AUDITOR: Data auditing in facial recognition systems," in *Proc. of USENIX Security*, 2023.

[51] K.-H. Chow, S. Hu, T. Huang, F. Ilhan, W. Wei, and L. Liu, "Diversity-driven privacy protection masks against unauthorized face recognition," in *Proc. of PETS*, 2024.

[52] B. Kawar, M. Elad, S. Ermon, and J. Song, "Denoising diffusion restoration models," in *Proc. of NeurIPS*, 2022.

[53] B. Kawar, G. Vaksman, and M. Elad, "Snips: Solving noisy inverse problems stochastically," in *Proc. of NeurIPS*, 2021.

[54] N. Carlini, J. Hayes, M. Nasr, M. Jagielski, V. Sehwag, F. Tramer, B. Balle, D. Ippolito, and E. Wallace, "Extracting training data from diffusion models," in *Proc. of USENIX Security*, 2023.

[55] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," in *Proc. of NeurIPS*, 2020.

[56] M. Tan and Q. Le, "Efficientnetv2: Smaller models and faster training," in *Proc. of ICML*, 2021.

[57] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *Proc. of CVPR*, 2009.

[58] A. Geitgey, "Face recognition," https://github.com/ageitgey/face_recognition, 2018.

[59] G. B. Huang, M. Mattar, T. Berg, and E. Learned-Miller, "Labeled faces in the wild: A database forstudying face recognition in unconstrained environments," in *Proceedings of the Workshop on faces in'Real-Life'Images: detection, alignment, and recognition*, 2008.

[60] T. Moon, M. Choi, G. Lee, J.-W. Ha, and J. Lee, "Fine-tuning diffusion models with limited data," in *Proc. of NeurIPS Workshop on Score-Based Methods*, 2022.

[61] I. Kemelmacher-Shlizerman, S. M. Seitz, D. Miller, and E. Brossard, "The megaface benchmark: 1 million faces for recognition at scale," in *Proc. of CVPR*, 2016.

[62] A. Nech and I. Kemelmacher-Shlizerman, "Level playing field for million scale face recognition," in *Proc. of CVPR*, 2017.

[63] A. Hore and D. Ziou, "Image quality metrics: Psnr vs. ssim," in *Proc. of ICPR*, 2010.

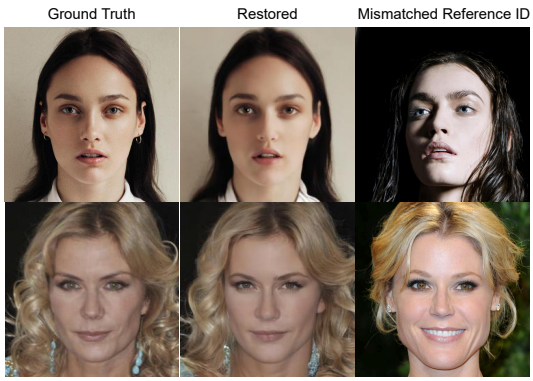

Figure 11: Examples of identity retrieval errors. The restored image (middle) from M_B is mismatched with a wrong identity in the reference database (right). In these examples, the restored faces are of high quality. Even though they are matched to a wrong identity (based on CelebA-HQ's identity labels), the mismatched faces also look similar.

[64] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, "The unreasonable effectiveness of deep features as a perceptual metric," in *Proc. of CVPR*, 2018.

[65] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE transactions on image processing*, vol. 13, no. 4, pp. 600–612, 2004.

[66] Z. Wang, J. Zhang, T. Chen, W. Wang, and P. Luo, "Restoreformer++: Towards real-world blind face restoration from undegraded key-value pairs," *IEEE Transactions on Pattern Analysis & Machine Intelligence*, vol. 45, no. 12, pp. 15 462–15 476, 2023.

[67] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter, "Gans trained by a two time-scale update rule converge to a local nash equilibrium," in *Proc. of NeurIPS*, 2017.

[68] A. M. Matwyshyn, A. Cui, A. D. Keromytis, and S. J. Stolfo, "Ethics in security vulnerability research," *IEEE Security & Privacy*, vol. 8, no. 2, pp. 67–72, 2010.

[69] A. Happe and J. Cito, "Understanding hackers' work: An empirical study of offensive security practitioners," in *Proc. of ESEC/FSE*, 2023.

[70] X. Qiu, C. Han, Z. Zhang, B. Li, T. Guo, and X. Nie, "Diffbfr: Bootstrapping diffusion model for blind face restoration," in *Proc. of MM*, 2023.

[71] T. Li and L. Lin, "Anonymousnet: Natural face de-identification with measurable privacy," in *Proc. of CVPR Workshops*, 2019.

[72] J. Cao, B. Liu, Y. Wen, R. Xie, and L. Song, "Personalized and invertible face de-identification by disentangled identity information manipulation," in *Proc. of ICCV*, 2021.

[73] Y. Wen, B. Liu, J. Cao, R. Xie, and L. Song, "Divide and conquer: a two-step method for high quality face de-identification with model explainability," in *Proc. of ICCV*, 2023.

Appendix A.
Case Study: Identity Retrieval Errors

As discussed in §5.3, our method achieves an identity retrieval accuracy of 95.9% under the heavy-blur setting. Here, we analyze the error cases. We find that most errors are due to the inherent challenges of facial recognition between identities that look similar. Figure 11 presents two example

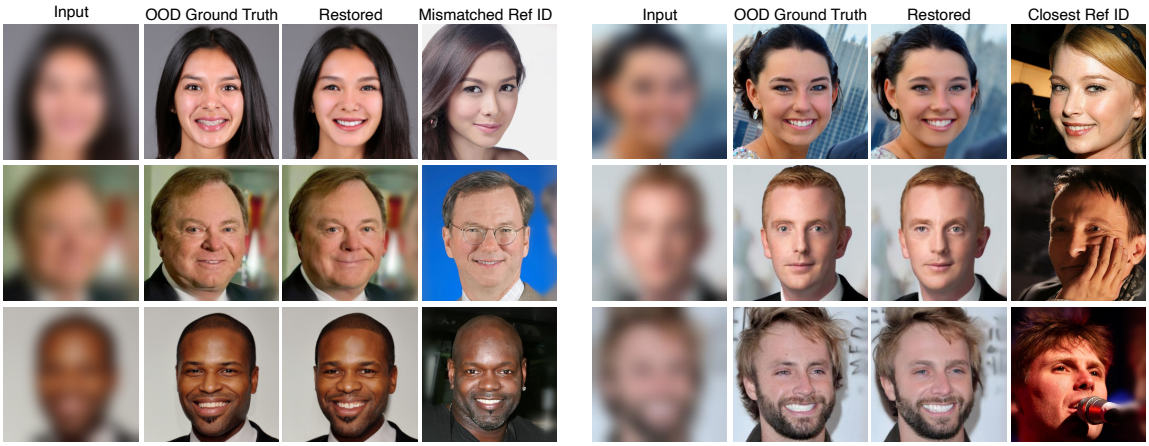

Figure 12: Face restoration examples for out-of-distribution (OOD) identities. (a) shows OOD identities missed by our detection method and (b) shows those detected correctly. For all of the examples, face restoration has been successful as the restored faces look similar to the original ground truth.

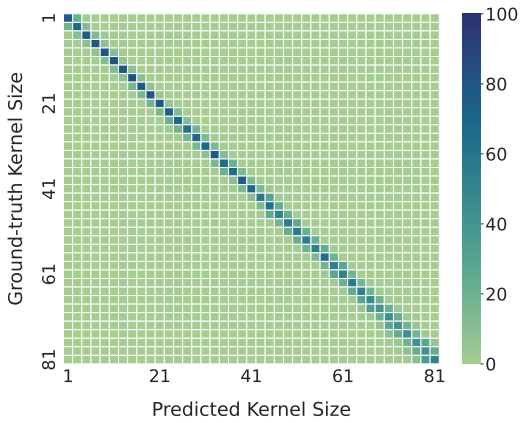

Figure 13: The confusion matrix of the kernel size estimator. The dark squares are aligned with the diagonal line, indicating that the model is highly accurate in predicting the kernel size of the input blurred images. The mean absolute error (MAE) is only 0.934.

error cases. For each case, we show their ground-truth image before blurring (left), the restored image (middle), and the matched reference images from the reference database D (right). We observe that the restored image looks reasonably similar to the ground-truth one. However, the restored image is incorrectly matched to a wrong identity in the reference database. This is because one of the face images of this identity (the right column) looks very similar to the restored image. This is an inherent limitation of face recognition within a large reference database (recall that our reference database contains 28,000 images and 6,084 identities).

Appendix B. Extra Example Images

Extra examples of out-of-distribution (OOD) identities are shown in Figure 12.

Appendix C. Kernel Size and Transferability

In this section, we present additional evaluation results for the kernel size estimation model, and the experiments to assess the model transferability against different Gaussian blur configurations.

Kernel Size Estimator Performance. Figure 13 presents the confusion matrix of the Gaussian kernel size estimator. The x-axis shows the predicted kernel size by the model, and the y-axis shows the ground-truth kernel size used to blur the input image. We can observe the prediction is highly accurate with prediction results aligning with the diagonal line of the matrix. The mean absolute error (MAE) of kernel size estimation is 0.934.

Transferability: Mismatched Kernel Size. To understand the impact of the mismatched kernel size on our method, we test our base model (trained on kernel size K =81) with images blurred with different kernel sizes. More specifically, we vary the kernel size from 71 to 91 to blur the input images to create the mismatch and then let the base model perform face restoration on these images. Due to computing resource limitations, we only run one round of face restoration per image for 50 sampled images ($n = 1$). Figure 14 shows example images restored by our model. We also present the quantitative metrics to assess the image quality and fidelity in Figure 15. In Figure 15, the y-range is set based on the metric values of the ground-truth images and those of the blurred images, which represent

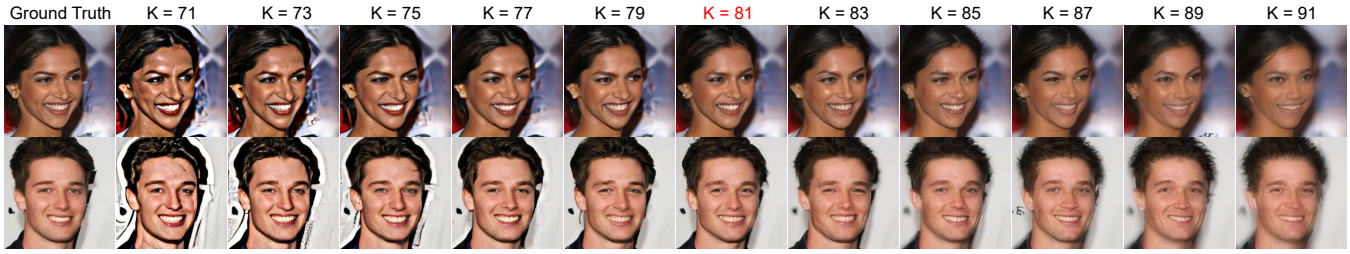

Figure 14: Example face restoration results with mismatched kernel sizes for Gaussian blur. The model is trained with $K = 81$. The testing image is blurred using a kernel size K varying from 71 to 91.

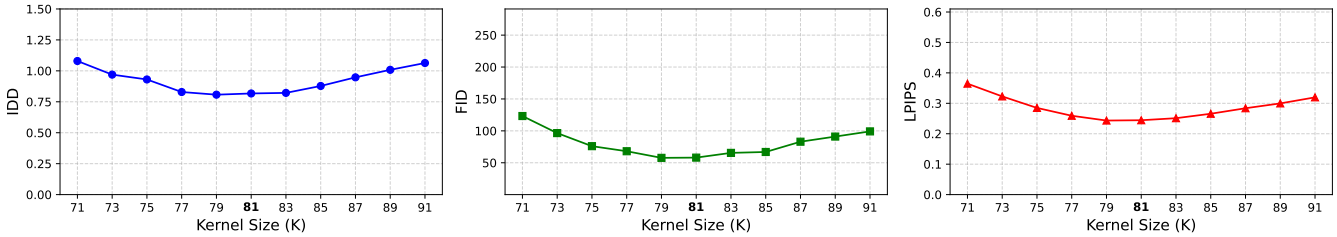

Figure 15: Image restoration quality and fidelity under mismatched kernel sizes. The ground-truth kernel size is $K = 81$. Combining with Figure 14, we show the model can tolerate mismatched kernel size with an offset of 6.

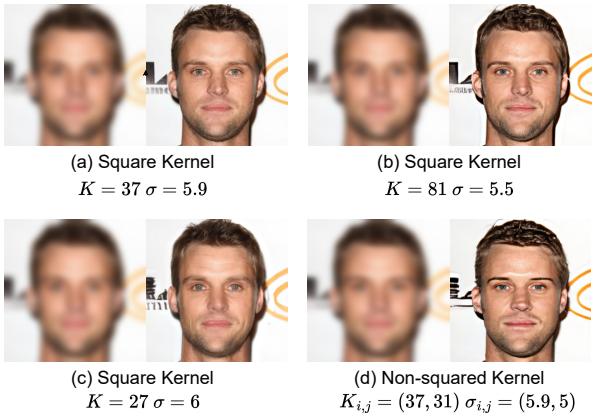

Figure 16: Examples of the transferability experiment results. Each subfigure shows a blurred image (left) and the restored image by our model (right). (a) shows the setting where we use the default K-σ dependency function in the existing Gaussian blur implementation; (b) and (c) show settings where we use a square kernel with different K-σ dependency functions; (d) shows the setting where we use a non-squared kernel (with a different K and σ on the two dimensions). All these settings have a similar blurring effect. The result shows that the restoration can still be successful under these settings.

the lower and upper bounds, respectively. The result shows that our model has some transferability over images blurred with mismatched kernel sizes. The restored faces still have a high-level resemblance compared with ground-truth images with a kernel size offset of 6. Recall that our kernel size estimator has an MAE lower than 1, which means that this level of mismatch is not a concern.

Transferability: Different $K - \sigma$ Dependency Functions. As mentioned in Section 2.1, most Gaussian blur implementations have a fixed dependency function between the kernel size K and the standard deviation σ, and thus attackers only need to predict K. We explore whether the model still works if future implementations change this dependency function. During these experiments, we have an interesting observation: regardless of how the $K - \sigma$ dependency changes, *as long as the blurring effect is similar to what* `Revelio` *is trained on, the system still works*. In practice, this means adversaries can use the kernel estimator to blindly predict a kernel size K, and select the corresponding M_B for face restoration for images blurred by an unknown $K - \sigma$ dependency function.

Figure 16 (a) shows the result from the default $K - \sigma$ dependency function in the existing Gaussian blur implementation ($K = 37$). Then in Figures 16 (b) and (c), we present the transferability experiments where we use different $K-\sigma$ dependency functions. We pick these settings because their blurring effect is similar to $K = 37$ under the old function (which is used to train M_B). This is determined by the trained kernel size estimator. The result shows that the restoration can still be successful under these settings. Taking Figure 16 (c) for example, we use $K = 27$ and $\sigma = 6$ to blur the image. Our kernel estimator predicts $K = 39$. This means our kernel estimator believes the blurring effect is similar to $K = 39$ *under the old function*. Then we choose M_B under the "light-blur" setting (which has been trained with $K = 37$, close to the predicted kernel size). We find the face restoration still works. The same observations also apply to Figure 16 (b).

Transferability: Non-Squared Kernel. Finally, we test the model transferability to a non-squared kernel. Here we use a kernel of a rectangle shape with different sizes and Gaussian distributions for the two dimensions. We have the same observation: as long as the blurring effect is similar to what `Revelio` is trained on, the system still works. We can still use the kernel size estimator (trained by the old square kernel) to predict the kernel size and select the corresponding M_B for face restoration. Figure 16 (d) demonstrates an example. The image is blurred by a non-squared kernel with $K = (37, 31)$ and $\sigma = (5.9, 5)$. The kernel estimator shows the blurring effect is similar to $K = 37$ under the old square kernel. In this case, the face restoration is still successful.

Appendix D. Defense Configurations

The configurations for the defense methods used in our experiments are as follows: (1) Rotation: we rotate each testing image by 180 degrees. (2) Gaussian Noise: the mean of Gaussian Noise is 0 and σ is 0.1. (3) JPEG Compression: according to the Pillow documents, when PIL images are saved to JPG or JPEG files, the image quality downgrades by 75% due to the JPEG compression algorithm. We use this configuration for our experiments. (4) Box Blur: the kernel size of the box blur is 31×31. This roughly matches the blurring effect of Gaussian blur with $K = 37$.