Robustness Analysis of Face Obscuration

Hanxiang Hao*, David Güera*, Amy R. Reibman†, Edward J. Delp*

* Video and Image Processing Lab (VIPER), Purdue University, West Lafayette, Indiana USA
† School of Electrical and Computer Engineering, Purdue University, West Lafayette, Indiana USA

Abstract

Face obscuration is often needed by law enforcement or mass media outlets to provide privacy protection. Sharing sensitive content where the obscuration or redaction technique may have failed to completely remove all identifiable traces can lead to life-threatening consequences. Hence, it is critical to be able to systematically measure the face obscuration performance of a given technique. In this paper we propose to measure the effectiveness of three obscuration techniques: Gaussian blurring, median blurring, and pixelation. We do so by identifying the redacted faces under two scenarios: classifying an obscured face into a group of identities and comparing the similarity of an obscured face with a clear face. Threat modeling is also considered to provide a vulnerability analysis for each studied obscuration technique. Based on our evaluation, we show that pixelation-based face obscuration approaches are the most effective.

1. Introduction

From TV news to Google StreetView, object obscuration has been used in many applications to provide privacy protection. Law enforcement agencies use obscuration techniques to avoid exposing the identities of bystanders or officers. To remove this identifiable information, Gaussian blurring or pixelation methods are commonly used. Median filtering is also used due to its simple implementation and its non-linearity, which translates to higher information distortion when compared to linear filters such as the Gaussian filter. These obscuration techniques are able to successfully prevent humans from recognizing the obscured objects. However, machine learning approaches can identify these objects using the subtle information left in the obscured images. In this paper, we focus on the performance analysis of the most common obscuration techniques for face redaction. Specifically, we study Gaussian blurring, median blurring, and pixelation to answer the following question: "How effective are these methods at concealing identity?"

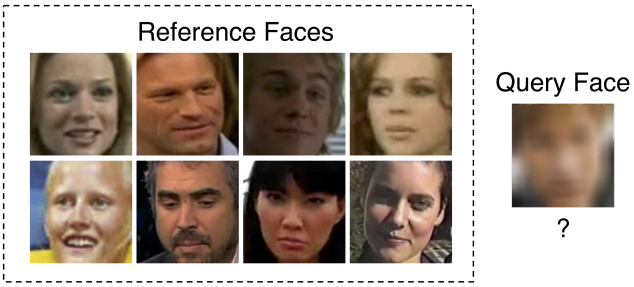


Figure 1: An illustration of the scenario studied in this paper. Given N identities with undistorted faces, we want to see if it is possible to determine the identity of the obscured query face.

Although these approaches are widely used by Internet news outlets, social media platforms, and government agencies, their performance has not been objectively measured. The lack of a formal study of these obscuration techniques makes them vulnerable to attacks. As shown by McPherson *et al*. [7], a deep learning model with a simple structure is able to identify individuals by analyzing their highly pixelated and blurred faces. This indicates that human perception is no longer the gold standard to examine the effectiveness of the obscuration methods. Therefore, we need to consider them under scenarios that allow us to see if we can extract the identifiable information from the obscured face, as shown in Figure 1. In particular, we want to test whether face identification is possible by comparing a distorted face against a set of undistorted reference faces and finding the closest match. In order to analyze the vulnerability of these obscuration methods, we design multiple threat models based on the attacker's knowledge of the obscuration method used. Our simplest threat model assumes that the attacker has no information of any obscuration methods.

In the most challenging threat scenario, we consider that the attacker knows the exact type of the obscuration method and its hyperparameters used. These unexplored threat models are necessary to offer a complete vulnerability analysis under realistic situations.

The main contributions of this paper are summarized as follows. First, we design a principal component analysis (PCA) based method to identify redacted faces under the face identification scenario. We also examine if deep learning based methods can be extended to the face verification scenario. Finally, we provide a comprehensive analysis of the obscuration performance of Gaussian blurring, median blurring, and pixelation under different threat models.

2. Related Work

Face Obscuration Methods. The goal of face obscuration is to remove all the identifiable facial information to prevent identification. As previously mentioned, Gaussian blurring and pixelation are frequently used in many commercial applications and social media sites. However, these techniques are not reliable. As we will show in Section 4, a poor choice for the kernel size of the Gaussian filter is not able to remove all identifiable information. An extreme resort to prevent information leaking is to simply gray out the entire facial region by setting all pixels in the facial area to a fixed value. However, this approach is rarely used because its visual effect is unpleasant, especially if there are many faces to be redacted. To overcome this issue, a variety of approaches have been proposed that try to balance the removal of identifiable information while preserving some facial features.

The first set of approaches, known as k-same methods [9, 4, 2], attempt to group faces into clusters based on personal attributes such as age, gender, or facial expression. Then, a template face for each cluster is generated. These methods are able to guarantee that any face recognition system cannot do better than $1/k$ in recognizing who a particular image corresponds to, where k is the minimum number of faces among all clusters [4]. In Newton *et al*. [9] and Gross *et al*. [4], they simply compute the average face for each cluster. Therefore, the obscured faces are blurry and cannot handle various facial poses. Du *et al*. [2] use the active appearance model [1] to learn the shape and appearance of faces. Then, they generate a template face for each cluster to produce obscured faces with better visual quality.

Generative adversarial network (GAN) methods [21, 17] are able to produce more realistic faces, since their discriminator is designed to guide the generator by distinguishing real faces from generated faces. Wu *et al*. [21] propose a model that generates an obscured face directly from the original face based on conditional adversarial networks [8]. A contrastive loss is used to enforce that the obscured face is different from the input face and a structure similarity loss is used to maintain the correspondence between the two faces. However, because they need to directly input the original faces, their obscuration performance is not guaranteed. To overcome this, Sun *et al*. [17] propose a model that is able to generate an obscured head without the original face region.

Privacy Analysis of Obscuration Methods. As mentioned in Section 1, although Gaussian blurring and pixelation are widely used, these methods might still leak sensitive information. Dufaux and Ebrahimi, and Sah *et al*. [3, 13] provide an analysis of the obscuration performance of common face identifiers and show the ineffectiveness of current obscuration methods. By using a simple deep learning model, McPherson *et al*. [7] also show that obscured images still contain enough information to perform accurate identification. They uncover the identity of images obscured with Gaussian blurring, pixelation, and privacy-preserving photo sharing (P3) [12] methods. Oh *et al*. [10] also propose a semi-supervised model that is able to identify the face under large variations in pose.

To extend the previous literature [3, 13, 7, 10], we first consider the face identification scenario. By mapping faces to known identities under different threat models, we analyze the vulnerability of each obscuration method. However, the requirement of known identities weakens this kind of analysis, since query faces usually come from unknown identities. To overcome this, we also provide a threat analysis under a more challenging and realistic setup: the face verification scenario. Specifically, we want to measure the similarity of an unknown redacted face to clear target faces. Since it allows the exposure of unknown identities, this scenario is more realistic than the face identification scenario.

3. Proposed Method

To evaluate the performance of the obscuration methods, we introduce the threat scenarios and face identification models.

3.1. Threat Model

In our model, the attacker aims to identify the redacted faces based on the information still present in the obscured images. To be clear, we define attacker as a face recognition system that tries to reveal the identity of the obscured faces. We design three threat models, which vary on how much information about the used obscuration approach is available to the attacker.

- Threat model T_1 assumes the attacker has no information of any obscuration method, which means that the attacker is only able to learn the facial features used for identification from clear faces. During the testing phase, it needs to extract the facial features from the obscured faces without any prior knowledge.

Figure 2: Illustration of eigenfaces. A: Example of faces in the AT&T dataset [14]; B: 10 eigenfaces; C: The average face among all images.

- Threat model T_2 assumes the attacker is aware of some obscuration methods, but not the same method used in the testing phase. This model provides more information to the attacker, since different obscuration methods may share similarities in terms of identifying facial features.
- Threat model T_3 assumes the attacker knows the exact type of the obscuration method and its hyperparameters, like the kernel size of Gaussian blurring or the pixel size of pixelation. Compared to T_2, T_3 provides the attacker with more information to identify the obscured faces.

3.2. Obscured Face Identification

For the obscured face identification problem, we assume a fixed number of identities. We treat this identification problem as a classification problem where the number of classes is equal to the number of identities. Compared to the verification task, this identification problem is easier to solve. We can design a simple identifier to compare the performance of different obscuration methods.

PCA is used for reducing data dimensionality and one of its application is facial recognition (also known as "Eigenfaces") introduced by Turk *et al.* [18]. The eigenfaces representation develops a fixed linear basis for the facial appearance with low dimensionality. It represents any face with the vector of coefficients of the linear combination. Figure 2 shows an example of eigenfaces. We will briefly formulate the PCA approach as introduced by Turk *et al.* [18]. Then, we will present our model for the obscured face identification problem based on PCA.

We first denote the training images as $\boldsymbol{\Gamma_i} \in \mathbb{R}^N, \forall i \in [1, M]$ and $\boldsymbol{\Gamma} \in \mathbb{R}^{N \times M}$, where N is the dimension of image and M is the number of images in the training set.

The average face shown in Figure 2 is given by

$$\boldsymbol{\Psi} = \frac{1}{M} \sum_{i=1}^{M} \boldsymbol{\Gamma_i}.$$

The l-th eigenface (eigenvector) $\boldsymbol{u}_l$ can be obtained from the sample covariance matrix $\boldsymbol{C}$ which is defined as

$$\boldsymbol{C} = \frac{1}{M} \sum_{i=1}^{M} \boldsymbol{\Phi_i} \boldsymbol{\Phi_i}^T = \boldsymbol{A}\boldsymbol{A}^T,$$

where

$$\boldsymbol{\Phi_i} = \boldsymbol{\Gamma_i} - \boldsymbol{\Psi},$$
$$\boldsymbol{A} = \begin{bmatrix} \boldsymbol{\Phi_1} & \boldsymbol{\Phi_2} & \dots & \boldsymbol{\Phi_M} \end{bmatrix} \in \mathbb{R}^{N \times M}.$$

Since the covariance matrix $\boldsymbol{C}$ is in $\mathbb{R}^{N \times N}$, it is not practical to compute its eigenvectors directly. Instead, we compute the eigenvectors $\boldsymbol{\nu}_l$ of the matrix $\boldsymbol{A}^T\boldsymbol{A} \in \mathbb{R}^{M \times M}$ and then compute $\boldsymbol{u}_l$ based on the following relationship:

$$\boldsymbol{u}_l = \boldsymbol{A}\boldsymbol{\nu}_l. \tag{1}$$

Note that the proof of Equation 1 can be found in [18]. Afterwards, we can find a linear combination of these eigenfaces to approximately reconstruct a new face image by

$$\hat{\boldsymbol{\Phi}}_i = \sum_{l=1}^{K} \boldsymbol{u}_l^T \boldsymbol{\Phi_i} \boldsymbol{u}_l = \sum_{l=1}^{K} \omega_l \boldsymbol{u}_l,$$

where K is the number of eigenfaces we select ($K \leq M$) and we further denote the projection weights as

$$\boldsymbol{\Omega} = \begin{bmatrix} \omega_1 & \omega_2 & \dots & \omega_K \end{bmatrix}^T$$

We can then perform face identification given a query face using a classifier based on its projection weights $\boldsymbol{\Omega}$ from the PCA approach, which can be defined as

$$\hat{y} = \arg\max_{y \in Y} p(y|\boldsymbol{\Omega}),$$

where $Y = \{1, 2, \dots, I\}$ is the identity label set when the total number of identities is I. Since the number of identities cannot be infinite, this method can only be applied to the obscured face identification problem. In our paper, we use a multi-layer perceptron (MLP) with one hidden layer as the face classifier.

3.3. Obscured Face Verification

The obscured face verification problem is defined as: given an obscured face and a clear face, decide if the two faces come from the same person or not.

In order to solve this verification problem, we project the image vector $\mathbf{x} \in \mathbb{R}^N$ to a lower-dimension latent vector in $\mathbb{R}^D$, where faces from the same person are closer together than faces from different people. Based on this idea, Schroff *et al.* [15] design a deep learning model, known as *FaceNet*, directly learning the mapping function transferring from the image space $\mathbb{R}^N$ to the compact Euclidean space $\mathbb{R}^D$. The distance from each point in the Euclidean

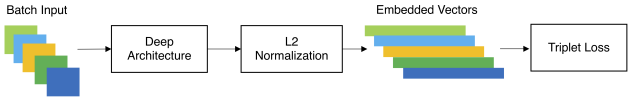


Figure 3: FaceNet model pipeline in training phase, modified based on Schroff *et al*. [15].

space represents the facial similarity. The model is trained with the *triplet loss*, which enforces a margin between the faces from the same identity and other faces from different identities in the Euclidean space. Define the mapping function as $f : \mathbb{R}^N \rightarrow \mathbb{R}^D$ and given an image $\mathbf{x} \in \mathbb{R}^N$, the corresponding projected point in the Euclidean space is $f(\mathbf{x}) \in \mathbb{R}^D$. Denote the *anchor* face as $\mathbf{x}^a$, *positive* face as $\mathbf{x}^p$ (with the same identity as $\mathbf{x}^a$) and *negative* face as $\mathbf{x}^n$ (with a different identity than $\mathbf{x}^a$). The objective of the triplet loss can be formulated as

$$\|f(\mathbf{x}^a) - (\mathbf{x}^p)\|_2^2 + \alpha < \|f(\mathbf{x}^a) - (\mathbf{x}^n)\|_2^2, \qquad (2)$$

where α is a predefined margin. By rearranging the terms in Equation 2, we can define the triplet loss function as

$$\mathcal{L}_t(\mathbf{x}^a, \mathbf{x}^p, \mathbf{x}^n) = \left[\|f(\mathbf{x}^a) - (\mathbf{x}^p)\|_2^2 - \|f(\mathbf{x}^a) - (\mathbf{x}^n)\|_2^2 + \alpha\right]_+,$$

where $[*]_+$ is the function that clips all negative values to 0. Selecting challenging triplets is crucial for fast convergence. Therefore, instead of randomly sampling the valid triplets, as proposed by Hermans *et al*. [5], we input a batch of images with multiple identities into the model and select all valid triplets to compute the triplet loss. The valid triplet has three faces with two of them coming from the same person and the surplus one coming from the different identity. The workflow of the FaceNet model in the training phase is shown in Figure 3, which inputs a batch of faces, computes the encoded vectors and calculates the triplet loss to train the model.

With the knowledge of triplet loss in hand, we can design our model as shown in Figure 4. The objective for our model is to ensure that the distance from the obscured face to the clear face from the same identity is closer than the distance between two different identities. We use two separate deep learning models with the same structure to extract the facial features from the obscured faces $\mathbf{x}_o$ and clear faces $\mathbf{x}_c$, which are formulated as $f_o : \mathbb{R}^N \rightarrow \mathbb{R}^D$ and $f_c : \mathbb{R}^N \rightarrow \mathbb{R}^D$. Therefore, the projected obscured face in the Euclidean space is $f_o(\mathbf{x_o})$, while the projected clear face in the Euclidean space is $f_c(\mathbf{x_c})$. We can further simplify the notation of the triplet loss function as follows. Use $\mathcal{L}_t(\mathbf{x}_o, \mathbf{x}_o)$ for the loss of all valid triplets from the obscured faces, use $\mathcal{L}_t(\mathbf{x}_c, \mathbf{x}_c)$ for the loss of all valid triplets from clear faces and use $\mathcal{L}_t(\mathbf{x}_c, \mathbf{x}_o)$ for the loss of all valid triplets of the combination of obscured and clear faces. Then, we can define the loss function of this obscured face identification model as

$$\mathcal{L} = \mathcal{L}_t(\mathbf{x}_o, \mathbf{x}_o) + \mathcal{L}_t(\mathbf{x}_c, \mathbf{x}_c) + \mathcal{L}_t(\mathbf{x}_c, \mathbf{x}_o),$$

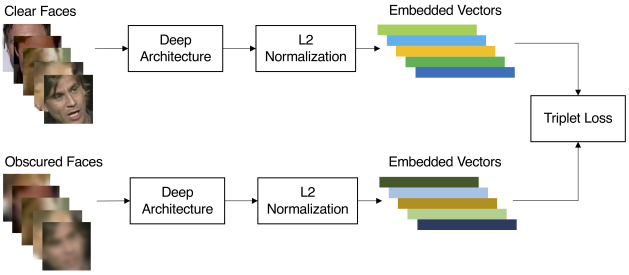


Figure 4: Obscured face verification model pipeline in training phase.

For the inference phase, we use obscured face and a clear face as inputs to the two mapping functions, f_o and f_c and compare the L_2 distance given the predefined threshold value to determine if the two faces come from the same person or not. The distance threshold value can be obtained based on the value that maximizes the identification accuracy of the validation set.

4. Experiments

In this section, we describe our experiments. First, we design a set of experiments for the identification task to provide a quantified analysis of different obscuration methods. Then, we extend them to the verification problem using the deep learning model to test these obscuration methods in this more realistic scenario.

4.1. Datasets

For the identification experiments, we use two datasets: the AT&T dataset [14] and the labeled face in the wild (LFW) dataset [6] and for the verification experiments, we use the YouTube face dataset [20]. The AT&T dataset provides 400 images of size 92×112 from 40 identities under different lighting conditions and facial expressions. We choose this dataset in order to compare the results from previous work [7, 13]. Moreover, the LFW dataset contains 13,000 images from 1,680 identities collected from the internet, and compared to the AT&T dataset it is more challenging in terms of scales and image quality. We use this dataset to provide a comprehensive threat analysis of the obscuration methods. Lastly, the YouTube face dataset contains 621,126 images of 1,595 identities (about 389 images per identity) which is much larger than the LFW dataset.

4.2. Obscured Face Identification Experiment

Experimental Design. We design several experiments to quantify the performance of the three obscuration methods: Gaussian blurring, pixelation and median blurring. In order to measure the identification accuracy, based on the design in Section 3.2, we train a PCA model with 150 eigenfaces and a 3-layer perceptron (the dimension of the hidden layer is 1024) as a classifier.

As mentioned in Section 4.1, we first train and test our model using the AT&T dataset to analyze the performance of Gaussian and median blurring with kernel sizes: 5, 15, 25 and 45, and pixelation with pixel sizes: 2, 4, 8, 16. Since there are 10 images for each identity, similar to [7, 13], we use 8 faces for training and 2 for testing. In order to compare the results, we use the top-1 accuracy as our metric.

We use the more challenging LFW dataset to provide a comprehensive analysis based on the aforementioned threat models. We remove the identities that have fewer than 25 faces in the dataset, which provides us with 2588 images from 42 people. We also convert the images into grayscale and use the Viola-Jones face detector [19] to detect faces. Afterward, we resize the cropped faces as 92×112, which is the same size as the AT&T dataset and choose the kernel (pixel) sizes as 5, 15, 25 and 45. In this experiment, we choose larger pixel sizes for the pixelation method than the experiments using the AT&T dataset to provide a more challenging situation. To evaluate the effectiveness of the obscuration methods, we use the cumulative match characteristic (CMC) curve as suggested by Dufaux and Ebrahimi [3] to compare the identification accuracy with respect to the identification rank.

We designed three experiments based on the three threat models to evaluate the obscuration methods. In the first experiment which is designed for T_1, the identifier is trained with the set of clear images and tested with obscured images. In the second experiment which is designed for T_2, the identifier is trained on both clear and obscured images and tested with the obscured images of the obscuration methods not used in the training set. In the third experiment which is designed for T_3, the identifier is trained on both clear and obscured images and tested with the obscured images employing the same obscuration methods.

Result. Table 1 shows the results of top-1 accuracy for face recognition on the different obscured images of the AT&T dataset. Note that McPherson *et al*. [7] used a deep learning model and Sah *et al*. [13] used a support vector machine classifier based on the features from the PCA approach. The *Original* column indicates that the deep learning approach yields the best identification accuracy, which shows that the CNN model is able to handle the facial features better than PCA-based approaches including ours and Sah *et al*. [13]. For the Gaussian blurring and median blurring, the results indicate that our model is able to identify the faces even when a significant amount of facial information has been obscured. As shown in Figure 5, when we increase the kernel size of the Gaussian and median filter, the facial features such as eyes or noses no longer exist in the images. Since there are only 10 images for each identity and only two of them are used for testing, the identifier is still able to identify the highly obscured face only from its remaining boundary of the facial region. For the pixelation method, the results from McPherson *et al*. [7] yield the best identification accuracy with the increasing pixel size. This shows that the CNN model is able to identify the faces even without noticeable facial features because of the small scale of the dataset.

Figure 5: Example of the obscured images of the AT&T dataset with the resolution of 92×112 (from left to right and top to bottom): Gaussian blurring with kernel size 5, 15, 25, 45; Median blurring with kernel size 5, 15, 25, 45; Pixelation with pixel size 2, 4, 8, 16.

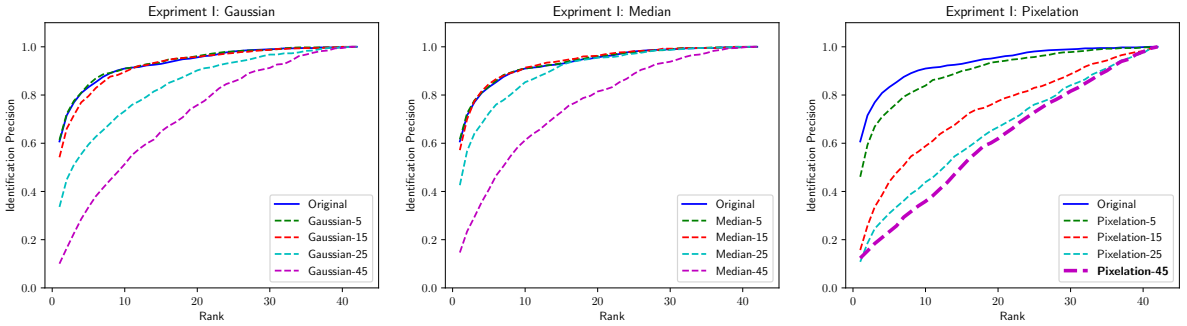
The pixelation results from our model show that it has a better obscuration performance as the pixel size increases. This is because pixelation is not only able to remove the facial features, but also the boundary of the facial region. Comparing the results from pixelation to the two blurring methods, we can see that the pixelation method is able to remove more identifiable information than Gaussian and median blurring.

Figure 6 shows the results for our PCA method for the three experiments with different obscuration methods and different kernel sizes for the LFW dataset. As a baseline, the results in Figure 6a indicate that without any obscuration the rank-1 and rank-5 accuracies are about 62% and 85%, respectively, based on the curve marked as *Original*.

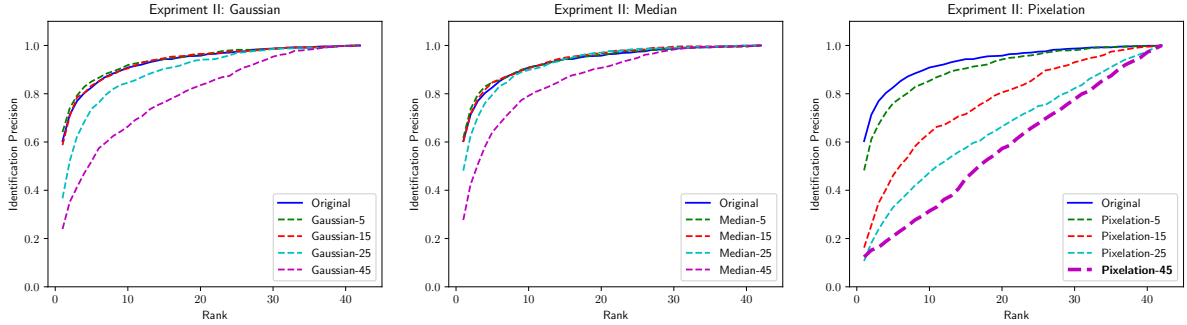
For treat model T_1 in experiment I, comparing the curves with different kernel sizes, all three methods have a better obscuration performance (*i.e*., lower identification precision) as the kernel size increases. Comparing the curves of different methods, the Gaussian and median filters with a kernel size of 5 almost have no difference from the results without any obscuration in the testing set. As shown in Figure 5 their blurring effect is really trivial. The pixelation method yields the best obscuration performance, especially

Method	Original	Gaussian				Median				Pixelation			
		5x5	15x15	25x25	45x45	5x5	15x15	25x25	45x45	2x2	4x4	8x8	16x16
[7]	**95.00**	-	-	-	-	-	-	-	-	95.00	**96.25**	**95.00**	**96.25**
[13]	88.74	85.25	73.75	61.25	31.25	-	-	-	-	87.50	83.75	70.00	36.25
Ours	92.50	**92.50**	**91.25**	**83.75**	**91.25**	**90.00**	**95.00**	**86.25**	**86.25**	**96.25**	83.75	79.00	60.00

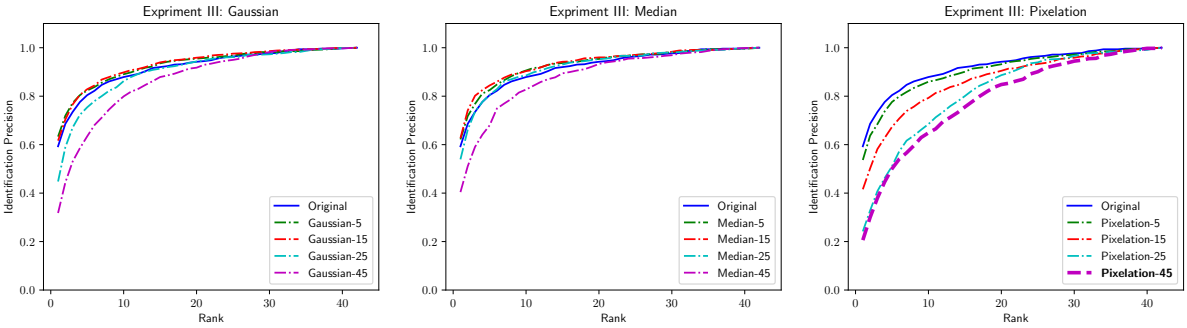
Table 1: Obscured face identification top-1 accuracy with the AT&T dataset.



(a) CMC plots of experiment I.



(b) CMC plots of experiment II.



(c) CMC plots of experiment III.

Figure 6: CMC plots for obscured face identification. The bold curve indicates the best obscuration method for each experiment.

for the pixel size of 45. Moreover, the Gaussian and median filters with a kernel size of 45 also have a good performance, but it is still worse than the pixelation method with a pixel size of 15, 25 and 45.

For treat model T_2 in experiment II, after adding the obscured images to the training set the identification precision of Gaussian and median filters increases, while the results of the pixelation method almost remain the same. This is caused by the pixelation with a large pixel size, which not only removes the identifiable facial features like eyes and

mouth, but also the outline of the face as mentioned in the previous experiment. Although the Gaussian and median filters can also obscure the identifiable information, the remaining facial outline provides the classifier information for identification.

For treat model T_3 in experiment III, if we train with the obscured images with the same obscuration operation used in the testing set, the obscuration performance for all methods will decrease. This can be especially seen for the pixelation method with a pixel size of 45, although it still achieves the best performance.

In summary, based on the results of obscured face identification, the pixelation method achieves the best obscuration performance compared to the Gaussian and median blurring methods.

4.3. Obscured Face Verification Experiment

Experimental Design. As previously mentioned, the deep learning method [7] yields the best performance even with the biggest pixel size. However, since the AT&T dataset experiment only has two images per identity for testing, the performance could be different when dealing with a larger dataset. Therefore, in order to further examine the obscuration performance of a larger dataset, we design a verification experiment. As mentioned in Section 4.1, we use the YouTube face dataset for this verification experiment and we split the dataset into training, validation, and testing with ratios of 0.6, 0.2 and 0.2 with cropped faces of 112×112. We also do random rotation ranging from $[-30°, 30°]$ for data augmentation to create more variation of facial poses. As suggested by Hermans *et al*. [5], we empirically choose a batch size of 128 with 4 images per identity. For the deep learning structure, we choose the VGG [16] model with output dimension 128 for both the obscured mapping function f_o and the clear mapping function f_c, since this model has been successfully implemented for facial recognition [11]. For the experiment design, we continue the third experiment (threat model T_3) in Section 4.2, which assumes that we know the obscuration type and train/test on the same obscuration method, including Gaussian blurring, median blurring, and pixelation. As shown in Figure 7, we still choose the same kernel/pixel sizes as in the previous identification experiments (5, 15, 25 and 45). During the testing phase, we sample face pairs from the same identity and different identities evenly. For the performance metric, since the face verification problem is just a binary classification problem, we choose the receiver operating characteristic (ROC) curve to examine the performance and use its area under the curve (AUC) as the numerical metric.

Result. As shown in Figure 8, similar to the results in Section 4.2, with the increase of kernel/pixel size, the three obscuration methods have a better obscuration performance. Observing the different obscuration methods with the same kernel/pixel sizes, the pixelation method yields the best obscuration performance, which confirms the results from Section 4.2. The performance of pixelation with a pixel size of 5 is even better than the Gaussian and median blurring with a kernel size of 15. Based on the visual effect from Figure 7, the abrupt edges generated from the pixelation method remove more identifiable facial information. Comparing the results of Gaussian and median blurring, the median blurring with a kernel size of 45 is slightly better than Gaussian blurring, since the facial details (eye region or nose) are completely removed when the kernel size increases.

Figure 7: Example of the obscured images of YouTube face dataset with the resolution of 112×112 (from left to right and top to bottom): Gaussian blurring with kernel size 5, 15, 25, 45; Median blurring with kernel size 5, 15, 25, 45; Pixelation with pixel size 5, 15, 25, 45.

As shown in Figure 9, we also produce visualization results to illustrate the performance of each obscuration methods in a more realistic scenario. We pick 6 clear faces from the YouTube face dataset (as shown on the right side of each image in Figure 9) and select an obscured image from the testing set. We intentionally add several challenging reference faces among the 6 faces in order to create extra difficulty for the model, like the first, fourth and fifth faces. The bar next to each clear face indicates the distance between the clear face and the obscured face to its immediate left and the highlighted green bar represents the face with the minimum distance. Note that the correct match is the second face from the top. Therefore, our model can correctly detect the obscured face from the Gaussian and median blurring with a kernel size of 15 (as shown in Figure 9a and 9b) and fails on the pixelation method with a pixel size of 5 (as shown in Figure 9c).

In summary, based on the results, the pixelation method achieves the best obscuration performance, which is similar to the results in the identification experiments.

5. Conclusion

In this paper, we provide a comprehensive analysis of three obscuration methods: Gaussian blurring, median blurring, and pixelation. We design a set of experiments to ex-

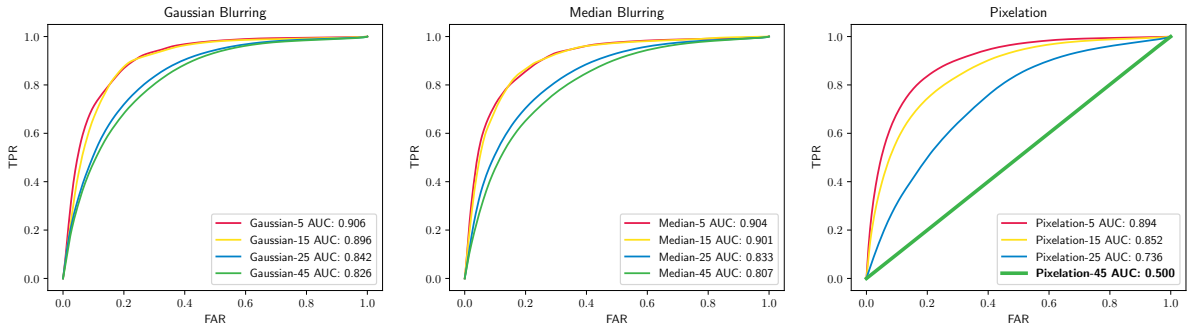


Figure 8: ROC curve for the experiments of obscured face verification. The bold curve indicates the best performance of this experiment.

(a) Gaussian with kernel size of 15.

(b) Median with pixel size of 15.

(c) Pixelation with pixel size of 5.

Figure 9: Obscured face verification for different obscuration methods. The 6 small images on the right side of each figure are the clear faces. The bar next to each image is the Euclidean distance to the obscured face on the left (the green bar is the one with minimum distance). The correct clear face is the **second** face from top.

amine if we are able to identify the obscured faces using a PCA based method and a deep learning based method. Based on the experiment results, we show that the pixelation method achieves the best performance since it brings abrupt edges, which improve the effectiveness of the face obscuration. Obscured faces using Gaussian and median blurring with a big kernel size are still able to be identified by our methods, although they are already unrecognizable by a human. Therefore, the designers of redaction systems need to be aware of choosing an effective obscuration method like pixelation and cannot only rely on human perception to determine a successful obscuration.

References

[1] T. F. Cootes, G. J. Edwards, and C. J. Taylor. Active appearance models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(6):484–498, June 2001. 2

[2] L. Du, M. Yi, E. Blasch, and H. Ling. Garp-face: Balancing privacy protection and utility preservation in face de-identification. *Proceedings of the IEEE International Joint Conference on Biometrics*, pages 1–8, September 2014. Clearwater, FL. 2

[3] F. Dufaux and T. Ebrahimi. A framework for the validation of privacy protection solutions in video surveillance. *2010 IEEE International Conference on Multimedia and Expo*, pages 66–71, July 2010. Singapore, Singapore. 2, 5

[4] R. Gross, E. Airoldo, B. Malin, and L. Sweeney. Integrating utility into face de-identification. *Proceedings of the International Workshop on Privacy Enhancing Technologies*, pages 227–242, May 2005. Cavtat, Croatia. 2

[5] A. Hermans, L. Beyer, and B. Leibe. In defense of the triplet loss for person re-identification. *arXiv:1703.07737v4*, November 2017. 4, 7

[6] G. B. Huang, M. Ramesh, T. Berg, and E. Learned-Miller. Labeled faces in the wild: A database for studying face recognition in unconstrained environments. *Technical Report*, October 2007. University of Massachusetts, Amherst. 4

[7] R. McPherson, R. Shokri, and V. Shmatikov. Defeating image obfuscation with deep learning. *arXiv:1609.00408v2*, September 2016. 1, 2, 4, 5, 6, 7

[8] M. Mirza and S. Osindero. Conditional generative adversarial nets. *arXiv:1411.1784v1*, November 2014. 2

[9] E. M. Newton, L. Sweeney, and B. Malin. Preserving privacy by de-identifying face images. *IEEE Transactions on Knowledge and Data Engineering*, pages 232–243, Feburary 2005. 2

[10] S. J. Oh, R. Benenson, M. Fritz, and B. Schiele. Faceless person recognition: Privacy implications in social media. *Pro-*

ceedings of European Conference on Computer Vision, pages 19–35, January 2016. Amsterdam, The Netherlands. 2

[11] O. M. Parkhi, A. Vedaldi, and A. Zisserman. Deep face recognition. *Proceedings of British Machine Vision Conference*, pages 41.1–41.12, Sept. 2015. Swansea, UK. 7

[12] M.-R. Ra, R. Govindan, and A. Ortega. P3: Toward privacy-preserving photo sharing. *10th USENIX Symposium on Networked Systems Design and Implementation*, pages 515–528, April 2013. Lombard, IL. 2

[13] S. Sah, A. Shringi, R. Ptucha, A. M. Burry, and R. P. Loce. Video redaction: a survey and comparison of enabling technologies. *Journal of Electronic Imaging*, 26(5):1 – 14 – 14, July 2017. 2, 4, 5, 6

[14] F. Samaria and A. Harter. Parameterisation of a stochastic model for human face identification. *Proceedings of 2nd IEEE Workshop on Applications of Computer Vision*, pages 138–142, December 1994. Sarasota, USA. 3, 4

[15] F. Schroff, D. Kalenichenko, and J. Philbin. Facenet: A unified embedding for face recognition and clustering. *Proceedings of the IEEE Computer Vision and Pattern Recognition*, pages 815–823, June 2015. Boston, USA. 3, 4

[16] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *Proceedings of the International Conference on Learning Representations*, May 2015. San Diego, CA. 7

[17] Q. Sun, L. Ma, S. Joon Oh, L. V. Gool, B. Schiele, and M. Fritz. Natural and effective obfuscation by head inpainting. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5050–5059, June 2018. Salt Lake City, UT. 2

[18] M. Turk and A. Pentland. Eigenfaces for recognition. *Journal of Cognitive Neuroscience*, 3(1):71–86, 1991. 3

[19] P. Viola and M. J. Jones. Robust real-time face detection. *International Journal of Computer Vision*, 57(2):137–154, May 2004. 5

[20] L. Wolf, T. Hassner, and I. Maoz. Face recognition in unconstrained videos with matched background similarity. *IEEE Conference on Computer Vision and Pattern Recognition*, pages 529–534, June 2011. Colorado Springs, USA. 4

[21] Y. Wu, F. Yang, Y. Xu, and H. Ling. Privacy-protective-gan for privacy preserving face de-identification. *Journal of Computer Science and Technology*, pages 47–60, January 2019. Beijing, China. 2