InfoScrub: Towards Attribute Privacy by Targeted Obfuscation

Hui-Po Wang[1] Tribhuvanesh Orekondy[2] Mario Fritz[1]

[1]CISPA Helmholtz Center for Information Security, Germany
[2]Max Planck Institute for Informatics, Saarland Informatics Campus, Germany

Abstract

Personal photos of individuals when shared online, apart from exhibiting a myriad of memorable details, also reveals a wide range of private information and potentially entails privacy risks (e.g., online harassment, tracking). To mitigate such risks, it is crucial to study techniques that allow individuals to limit the private information leaked in visual data. We tackle this problem in a novel image obfuscation framework: to maximize entropy on inferences over targeted privacy attributes, while retaining image fidelity. We approach the problem based on an encoder-decoder style architecture, with two key novelties: (a) introducing a discriminator to perform bi-directional translation simultaneously from multiple unpaired domains; (b) predicting an image interpolation which maximizes uncertainty over a target set of attributes. We find our approach generates obfuscated images faithful to the original input images and additionally increases uncertainty by 6.2× (or up to 0.85 bits) over the non-obfuscated counterparts.

1. Introduction

A tremendous amount of personal visual data is shared on the internet everyday [14] e.g., camera photos shared on social networks. The wide range of private information inadvertently leaked as a consequence is severely underestimated [12]. To prevent catastrophic side-effects of such privacy leakage (e.g., online harassment, deanonymization), it is crucial to study techniques that allow users to limit the amount of private information revealed in images before they are shared online. To this end, we build upon recent advances in computer vision techniques and present methods to protect the privacy of individuals in visual data.

Specifically, we explore the notion of obfuscating selected privacy attributes in images. Most literature around obfuscating image regions focus on detection and masking (e.g., by blurring) a narrow set of privacy attributes – predominantly faces and license plates. However, a recent line of work [12, 11] extends these efforts to a much wider range of attributes, largely motivated that images contain various bits of information – much like pieces of a jigsaw puzzle – which in conjunction can compromise the individual's privacy. In this work, we study targeted obfuscation of a variety of privacy attributes (e.g., hair color, facial hair) in images, many of which cannot be clearly localized (e.g., age).

However, as a natural side-effect, obfuscation-based approaches towards manipulating images lead to destroying the 'utility' of images. Various concepts of utility were explored recently. The predominant notion [5, 2, 15] is to define utility w.r.t a complementary set of non-sensitive utility attributes that can be inferred from images e.g., emotion. As a result, such formulations treat obfuscation as a minimax game between inferences of disjoint privacy and utility attributes. However, in this work, we hope to capture the usefulness of an obfuscated image beyond a (typically small) set of categorical attributes. Consequently, our work considers the visual quality of the image as a proxy to the utility, which is inherently important for online photo sharing.

Our solution involves synthesizing images resembling the original input image, albeit with certain privacy attributes removed. The solution is reminiscent of a recent line of work of performing attribute manipulation on images using generative adversarial networks. However, as we will show later, attribute-manipulation GANs pose numerous subtle problems when the task involves manipulating *privacy* attributes. In particular: (i) they fail to associate non-removal of a particular attribute with a large (privacy) cost; (ii) manipulated images collapse to extreme solutions (attribute present or absent), whereas the required obfuscated solutions are typically in-between, i.e., exhibiting maximum entropy over presence/absence of the targeted attribute; and (iii) attribute manipulations fail to generalize to unseen adversaries. To tackle these challenges, we find existing attribute-manipulation GANs limited, and work towards attribute-*obfuscation* GANs.

We present a two-stage approach towards targeted obfuscation of privacy attributes in images. The first stage performs *attribute inversion* in images: given an input image, to toggle the presence/absence of the target attribute. We find existing image-manipulation techniques only partially invert the attributes, and hence fail to generalize to unseen adversaries. We tackle the partial inversion problem by employing a novel bi-directional discriminator and additionally employ adversarial training to update the discriminator. Consequently, we find the first stage of our approach more effective in inverting attributes than attribute-manipulation counterparts. In the second stage, we extend our approach to performing *attribute obfuscation* by maximizing uncertainty over the presence of the target attribute. The key challenge here is the lack of ground-truth examples containing obfuscated images to guide the supervision. To combat this, our second-stage model searches for the obfuscated image by interpolating the input image and the corresponding attribute-inverted image.

We evaluate our approach on CelebA by obfuscating ten facial attributes (e.g., gender, hair color), while keeping the generated image faithful to the original i.e., preserving the remaining privacy attributes and image fidelity. We highlight that our evaluation setting is more involved than existing obfuscation literature: (i) we consider a wider range of privacy attributes to obfuscate; and (ii) we forego a constrained and limited set of categorical utility attributes, and solely consider the broader notion of image fidelity. In this challenging setting, we find our approach successfully manipulate privacy attributes. For instance, we find our approach inverts presence and absence of privacy attributes, with 84.5% accuracy, an increase of 18.5% achieved by recent image-translation model such as StarGAN. Furthermore, apart from inverting privacy attributes, we find our approach equally capable of obfuscating them i.e., maximizing uncertainty of attribute predictions. Specifically, we observe an average increase of entropy from 0.2±0.21 bits to 0.81±0.18 bits (maximum entropy = 1 bit) across inferences over ten privacy attributes. Our results indicate we can significantly reduce the amount of private information leaked by an image, while retaining its faithfulness, and provide a viable privacy-preserving approach towards visual information sharing.

2. Related Work

Attribute Manipulation. Generative adversarial networks (GANs) [6, 9, 10] have been recently extended to edit visual attributes (e.g., changing emotions in faces) [4, 13, 8, 1] in images. Central to these methods is using an attribute (often referred to as 'domain') classifier to guide the editing process. While these works produce often produce photo-realistic images, they are trained to fool a fixed known 'adversary' (the attribute classifier). Consequently, we find they fail to generalize to new adversaries (unseen attribute classifiers). This is particularly problematic from a privacy stand-point, where one does not know before-hand the model used to infer attributes from images. To tackle the generalization issue, our proposed method first trains the classifier in an adversarial manner and adopts a proposed bi-path classifier to solve the confusion problem, which is discussed in detail in Section 3.2.

Privacy-Preserving Learning. In addition to attribute manipulation, several works propose to obfuscate private information from input images within a GAN-based formulation. To name a few, Bertran *et al*. [2] learn to modify images by incorporating competition between generators and proxies of adversaries into the training, encouraging generators to better conceal sensitive information. Similarly, Roy *et al*. [15] adopts a strategy to produce privacy-preserving embeddings. Creager *et al*. [5] first learn disentangled representations with TC-VAE [3] and supervision from attributes. During the test time, it hides sensitive information by disabling the corresponding position in representations. While these works are effective in concealing privacy attributes, they do not generate realistic images which violates the original intention of data sharing (e.g., shared across social media). Moreover, except Creager *et al*. [5], these algorithms need to define attributes in the training time and are not able to change privacy settings during the test time. In this paper, we propose a framework that provides users flexibility over a variety of sensitive attributes and obfuscate images while retaining image fidelity, which is crucial to enable photo sharing.

3. Method

In this section, we present the proposed image obfuscation framework which provides users flexibility to remove an arbitrary subset of sensitive attributes while retaining the fidelity of generated images. Before fleshing out the details in the remainder of this section, we first provide an overview of our two stage approach (shown in Fig. 1).

Stage I: Attribute Inversion. We first learn an image-to-image translation model that performs privacy attribute inversion: given an input input x, to synthesize an image $\bar{x}$ (faithful to x), but where the presence/absence of the target privacy attribute is toggled. The key ideas in our model involve: (i) training an encoder which disentangles the visual features in the image from the attribute information; (ii) manipulating the disentangled attribute information to signal inversion targets; and (iii) introducing a bi-directional discriminator, which we find crucial to alleviating issues of partial inversion of attributes. We remark that while our approach shares some similarities with attribute manipulation strategies [4, 13], we tackle specific challenges to better generalize to unseen attribute classifiers (critical when enforcing privacy), while producing photorealistic images.

Stage II: Attribute Obfuscation. We further extend our

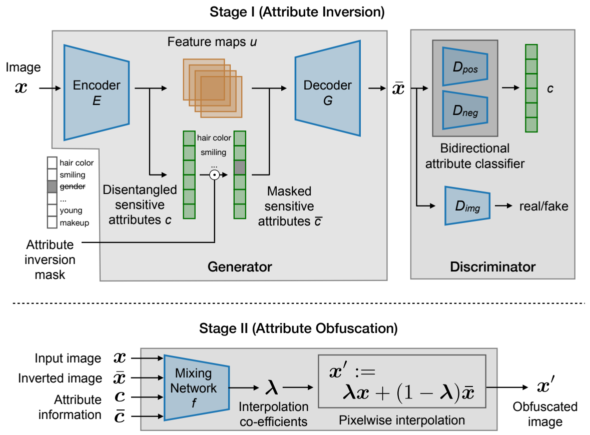


Figure 1. Our approach involves two stages: (I) we first invert the presence of the target attribute (e.g., gender) in the input image x to obtain $\bar{x}$, followed by (II) crafting an obfuscated image x' as an interpolation of x and $\bar{x}$ to exhibit maximum uncertainty over the target attribute.

approach to synthesize obfuscated images i.e., images with high uncertainty over presence of target attribute. As shown in the lower part of Fig. 1, we achieve obfuscation using a mixing network, which predicts the pixel-wise linear interpolation coefficients λ between the original input image x and the attribute-inverted image $\bar{x}$. Consequently, we arrive at an obfuscated but photorealstic image x', which displays high entropy over the target attribute.

Now, we move to discussing in detail the first- (Sections 3.1-3.3) and second-stage (Section 3.4) of our approach to perform image obfuscation.

3.1. Attribute Inversion (Stage I): Overview

The proposed approach transforms an input image $x \in R^{H \times W \times 3}$ to produce a complementary edited image $\bar{x}$ via an encoder-decoder architecture. To perform arbitrary attribute inversion during a single forward-pass, the approach allows manipulation on the disentangled code produced by the encoder. As shown in Figure 1, there are four sub-networks within the framework: an encoder E, a decoder G, a compound bi-directional attribute classifier $D_{\text{pos}}/D_{\text{neg}}$, and an image discriminator D_{img}. We now discuss each of these sub-networks.

Disentanglement via Encoder E. To infer and decouple the underlying information, the encoder E models the information by encoding input images

$$(u, c) = E(x), \tag{1}$$

where u is a set of feature maps describing non-sensitive information, and c is a vector describing sensitive attributes. The two representations are encouraged to be independent of each other. Using the specified attribute edits (via a binary encoded inversion mask), c is modified to $\bar{c}$, which is subsequently used to sanitize the input image.

Decoder G. With the disentangled representations, the decoder G models the residuals that change the pixels with high information leakage risks. Formally, we have

$$\bar{x} = x + G(u, \bar{c}). \tag{2}$$

The motivation behind the design is that most of pixels in the input image are unrelated to sensitive information. Therefore, we can lower the cost of learning image generation by simply modeling residuals.

Image- (D_{img}) and Attribute-level Discriminators. We consider two constraints on the generated image $\bar{x}$: (a) it should resemble realistic images; and (b) it should fool an attribute-level discriminator trained to classify privacy attributes. To tackle (a), we introduce an image-level discriminator D_{img} to synthesize realistic images in an adversarial manner [6]. We elaborate (b) in the next section, as naively introducing an attribute-level discriminator is problematic.

3.2. Bi-directional Discriminator

Common to many attribute manipulation techniques is employing an attribute-level discriminator (also referred to as domain classifier in literature) during training. This discriminator (which we refer to as D_{attr}) is trained only on non-generated real images and is used to provide the generator G feedback on whether the attribute was successfully manipulated. Directly employing D_{attr} for the problem of privacy attribute manipulation leads to two challenges. We describe the challenges and how we address them in the following paragraphs.

Discriminator Overfitting. We find training the attribute discriminator only on real images leads to over-fitting issues, in which the model solely learns to fool the discriminator by removing specific regions of target attributes (e.g., removing the bridge of eyeglasses to eliminate the activation). This increases the risk that sensitive information can be still recognized from the processed images and violates our goal of protecting visual privacy. To alleviate the problem, one common approach is utilizing adversarial training by updating the discriminator with generated images.

Partial Inversions using D_{attr}. Another key issue in performing attribute manipulation in our setting using D_{attr} arises from high-confidence predictions in low-density data regions. We find such discriminator feedback encourages the generator to sample partially inverted images in the low-density region where although the discriminator is correctly fooled, the presence/absence of the attribute is visually ambiguous due to proximity to the decision boundary. For instance, as shown in Fig. 2, the generated images x_2 and x_3 (close to vertical gray decision boundary) leads to high-confidence presence/absence predictions (of attribute 'blonde'), although both images are visually indistinguishable.

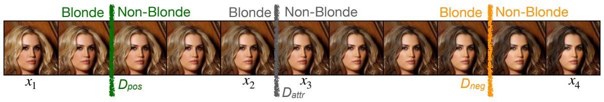
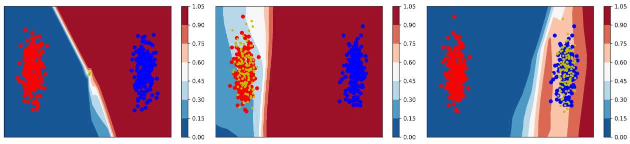


Figure 2. Attribute discriminator D_{attr} vs. our bi-directional discriminator $D_{\text{pos}} \cup D_{\text{neg}}$. Vertical lines illustrate decision boundaries between presence and absence of attribute 'blonde'. We find translating an input image (e.g., removing 'blonde' attribute in x_1) using only D_{attr} incorrectly encourages partially attribute-inverted images drawn close to the decision boundary (x_3). However, our discriminator learns a tighter decision boundary (D_{neg}) for this translation and produces an image (x_4) better representing inversion of the attribute. We find a similarly effective translation in the other direction as well (e.g., adding 'blonde' to x_4 producing x_1) using D_{pos}.



(a) D_{attr} (b) D_{pos} (c) D_{neg}

Figure 3. Two-Gaussian toy dataset. We translate points in one Gaussian to the other Gaussian and produce confidence maps by (a) a conventional classifier and a bi-directional classifier, where (b) is D_{pos} aiming for positive-to-negative translation and (c) is D_{neg} aiming for negative-to-positive translation. Red, blue, and yellow dots represent points in two Gaussian and translated points, respectively. Note that optimal generation often happens near the probability 0.5. Thus, an accurate boundary could benefit translation.

Bi-directional Discriminator. Our core idea to tackle the partial inversion generation problem is to encourage tighter decision boundaries around the positive and negative classes. We achieve this using a bi-directional discriminator composed of two attribute classifiers: D_{pos} (to identify positive→negative image translations) and D_{neg} (negative→positive). In Fig. 2 this is illustrated by the green and orange vertical lines; notice the decision boundaries are closer to high-density regions.

We additionally validate the effectiveness of our bi-directional discriminator over D_{attr} on a synthetic dataset composed of two Gaussians. As shown in Fig. 3, each Gaussian cluster represents positive/negative examples and the goal is to ideally perform axis-aligned bi-directional translation (red⇆blue) from one cluster to another. In Fig. 3(a), we see that using D_{attr} leads to translated samples (yellow points) to collapse in a low-density region near the decision boundary. Our discriminator (Fig. 3(b, c)) produces reasonable translations (where translated yellow points are now in high-density regions) aided by tighter decision boundaries.

We implement bi-directional discriminator using two classifiers D_{pos} and D_{neg} (also illustrated in Fig. 1), where D_{pos} judges positive-to-negative attribute inversion and D_{neg} judges negative-to-positive. We extend standard binary cross entropy loss ($\mathcal{L}_{\text{bce}}$) to a bi-directional loss ($\mathcal{L}_{\text{bi}}$) to satisfy our constraint:

$$\begin{aligned}\mathcal{L}_{\text{bi}}(x, y^{\text{org}}, y^{\text{tar}}) = & y^{\text{org}}\mathcal{L}_{\text{bce}}(D_{\text{pos}}, x, y^{\text{tar}}) \\ & + (1 - y^{\text{org}})\mathcal{L}_{\text{bce}}(D_{\text{neg}}, x, y^{\text{tar}}),\end{aligned} \tag{3}$$

where x is the input image, y^{org} and y^{tar} denote the original and target labels, and $D_{\cdot}$ is the classifier used to compute loss function. The overall objective function is realized as follows

$$\max_{G} \max_{D} \mathcal{L}_{\text{bi}}(x, y, y) + \mathcal{L}_{\text{bi}}(\bar{x}, y, \bar{y}), \tag{4}$$

where $\bar{x}$ is translated images produced by G along with the target labels $\bar{y}$. Thus, the bi-directional classifier can be smoothly applied to perform attribute inversion. In Figure 3(b, c) we observe that the bi-directional classifier can provide tighter decision boundaries for both directions and perform effective axis-aligned translations. Empirically, we find that the performance is sufficiently satisfactory when the discriminator and the bi-path classifier share the same feature extractor and only differ in the last few layers.

3.3. Learning to Invert Attributes

The proposed framework for the first stage of our approach is trained to minimize a weighted sum of loss functions which regularize the model to achieve our goal discussed in Section 3.1:

$$\mathcal{L}_G = \mathcal{L}_{\text{rec}} + \mathcal{L}_{\text{cclf}} + \mathcal{L}_{\text{bi}} + \mathcal{L}_{\text{adv}} + \lambda_1\mathcal{L}_{\text{util}} + \lambda_2\mathcal{L}_{\text{reg}}, \tag{5}$$

where λ_1 and λ_2 together controls the trade-off between utility and privacy. We introduce each loss functions in detail over the following paragraphs.

Reconstruction Loss. Given input images x, we train the encoder E to produce disentangled representations $(u, c) = E(x)$ to characterize attribute-independent visual features u and disentangled attribute representation c. We adopt an L_1 loss to enforce the reconstructed images $\hat{x} = G(u, c)$ to resemble input images x:

$$\mathcal{L}_{\text{rec}}(\hat{x}, x) = \|\hat{x} - x\|_1 . \tag{6}$$

With this process, we ensure the information contained in images are well-preserved.

Code Classification Loss. To encode the attribute information into c, a mean square loss is imposed on c. Therefore, each element c_i in c represents an binary attribute of x.

$$\mathcal{L}_{\text{cclf}}(c, y) = \|c - y\|_2^2 , \tag{7}$$

where y is the ground truth of sensitive attributes.

Apart from image reconstruction, the decoder generates attribute-inverted images $\bar{x} = G(u, \bar{c})$ given the modified sensitive code $\bar{c}$ along with the non-sensitive code u. In particular, we first create $\bar{c}$ by replacing certain elements of c with binary variables $s \in \{0,1\}^{N_s} \sim \text{Cat}(K = 2, p = 0.5)$ and define the modified label $\bar{y}$ as follows.

$$\bar{y}_i = \begin{cases} s & \text{if } c'_i \neq c_i \\ y_i, & \text{otherwise} \end{cases} \tag{8}$$

The number of inverted attributes N_s is determined by $\text{Cat}(K = \frac{N_A}{2}, p = \frac{2}{N_A})$, which provides the flexibility that the model can invert multiple attributes simultaneously. Note that during the test time, every element c_i can be arbitrary assigned to either 0 or 1.

Bi-Directional Attribute Loss. The attribute-inverted images $\bar{x}$ are required to fool the attribute classifiers in an adversarial manner. As motivated in Section 3.2, we apply the bi-directional loss to avoid the partial attribute inversion problem. For the generator, we force the generated images to align with $\bar{y}$.

$$\mathcal{L}_{\text{bi}}(\bar{x}, y, \bar{y}) = y\mathcal{L}_{\text{bce}}(D_{\text{pos}}, \bar{x}, \bar{y}) + (1-y)\mathcal{L}_{\text{bce}}(D_{\text{neg}}, \bar{x}, \bar{y}) \tag{9}$$

The above contrasts classifiers which are typically trained to recognize the original attributes, where m masks out positions that c_i is not edited:

$$\mathcal{L}_{\text{attr}}(\bar{x}, y) = m\cdot[y\mathcal{L}_{\text{bce}}(D_{\text{pos}}, \bar{x}, y)+(1-y)\mathcal{L}_{\text{bce}}(D_{\text{neg}}, \bar{x}, y)], \tag{10}$$

Image Adversarial Loss. In addition to fooling the attribute classifiers, we also impose image adversarial loss to encourage the realistic image generation. The intuition is that, without the constraint, the model could generate adversarial examples to fool the attribute classifiers, which violates our motivation.

$$\mathcal{L}_{\text{adv}}(\bar{x}, x) = \log D_{\text{img}}(x) + \log(1 - D_{\text{img}}(\bar{x})) \tag{11}$$

Content Regularization Loss. The attribute-inverted images should resemble the original images although some of attributes are modified. We additionally introduce cycle-consistent reconstruction to the model, encouraging the model to preserve the major content of the original images. We introduce the notion of margin to form hinge loss, which balances the tradeoff between privacy and content distortion.

$$\mathcal{L}_{\text{reg}}(\bar{x}, x) = \max(\|E(\bar{x}) - E(x)\|_1 - \delta_1, 0), \tag{12}$$

where δ_1 indicates tolerance of content distortion.

Utility Loss. In addition to preserving content, non-target attributes, namely those with unchanged c_i, also need to be preserved. We ensure it by classical binary cross entropy loss. Similarly, the loss function is controlled by the margins. Note that we impose the loss function on both reconstructed and sanitized images to facilitate the learning.

$$\mathcal{L}_{\text{util}} = \max(\mathcal{L}_{\text{bi}}(\bar{x}, y) - \delta_2, 0), + \max(\mathcal{L}_{\text{bi}}(\hat{x}, y) - \delta_3, 0), \tag{13}$$

where δ_2 and δ_3 indicate tolerance of attribute distortion for the sanitized and reconstructed images, respectively. The margin δ_3 is often set to be zero since attributes of reconstructed images are unchanged.

3.4. Attribute Obfuscation (Stage II)

With our model (Fig. 1) trained to minimize loss terms (Eq. 5), we are equipped to *invert* attributes i.e., perform bi-directional translations by manipulating presence and absence of targeted attributes in an input. Now, we extend the approach to *obfuscate* the image i.e., introduce *uncertainty* over targeted attributes. To achieve obfuscation, given an input image x, we first generate its complement $\bar{x}$ by inverting the presence of the target attribute. We then generate the obfuscated image x' as a linear interpolation between x and $\bar{x}$:

$$x' = I_{\text{mix}}(x, \bar{x}, \lambda) = \lambda x + (1-\lambda)\bar{x}, \tag{14}$$

where the mixing coefficient $\lambda \in [0,1]^{H\times W}$ is generated to maximize the prediction uncertainty with respect to the target attribute.

We train a network f to predict image-specific mixing coefficients λ:

$$\lambda = f(x, \bar{x}, c, \bar{c}), \tag{15}$$

where x is the input image, $\bar{x}$ is the attribute-inverted image, c is the sensitive code, and $\bar{c}$ is the modified sensitive code. We model f using 5 residual blocks followed by a 1×1 filter. The network is trained to produce coefficients that lead to obfuscated images with maximum uncertainty preserving photorealism:

$$\mathcal{L}_{\text{ent}}(x', y') = \mathcal{L}_{\text{adv}}(x') + \mathcal{L}_{\text{bi}}(x', y'), \tag{16}$$

where $\mathcal{L}_{\text{adv}}$ encourages x' to be realistic and $\mathcal{L}_{\text{bi}}$ encourages the interpolated images x' to have maximum entropy with respect to the prediction of both D_{pos} and D_{neg} with labels y' (with $p(y'_i)$ set to 0.5 for target attribute i).

4. Experiments

4.1. Setup

Dataset. CelebA is composed of 200K human face images associated with 40 attributes. We choose a subset of 10 disjoint attributes, that is representative of sensitive information. Every input image is center-cropped by 178x178 and then resized to the resolution 128x128. We use 150K images sorted by indices as training data and form a balanced dataset for evaluation from the remaining 53K images.

		Blonde hair	Eyeglass	Heavy makeup	Male	No beard	Wavy hair	Wearing lipstick	Young
TPR (%)	Real	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0
	IcGAN [13]	61.6	12.1	39.6	43.8	97.5	53.1	46.3	75.6
	StarGAN [4]	10.2	2.5	65.1	41.9	51.3	36.6	65.5	**22.8**
	Ours	**6.0**	**1.7**	**21.0**	**4.3**	**10.8**	**28.6**	**13.7**	30.1
TNR (%)	Real	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0
	IcGAN [13]	31.2	84.3	85.6	63.7	**5.69**	65.2	79.8	37.7
	StarGAN [4]	22.7	7.5	74.5	36.7	17.2	41.4	80.3	**25.0**
	Ours	**14.3**	**5.5**	**8.0**	**3.3**	15.3	**28.6**	**9.3**	29.3
Acc (%)	Real	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0
	IcGAN [13]	35.6	79.5	67.4	55.6	82.6	61.3	63.7	66.5
	StarGAN [4]	16.4	5.0	69.8	39.3	34.3	39.0	72.9	**23.9**
	Ours	**10.2**	**3.6**	**14.5**	**3.7**	**13.1**	**28.6**	**11.5**	29.7

Table 1. Quantitative results for attribute inversion. Lower (adversary) scores are better.

Modeling the Adversary. To fairly compare our method to prior works, we train a ResNet-18 [7] classifier on the same training data, which acts as an adversary attempting to infer privacy attributes from images. The adversary ResNet classifier used during evaluation is: (i) trained independently to our method; and (ii) significantly more complex than the attribute discriminators in our method. Furthermore, as this attribute classifier achieves near-perfect attribute prediction accuracy, we argue it models a strong unseen adversary to evaluate obfuscation techniques.

4.2. Evaluation Metrics

We now present evaluation metrics for both stages of our approach: stage 1 (which *inverts* the target attribute) and stage 2 (which *obfuscates* i.e., maximizes uncertainty of the target attribute).

Evaluating Attribute Inversions. We consider the following metrics: (i) True Positive Rate (TPR = TP / (TP + FN)): to evaluate how well we are able to 'remove' the target attribute; (ii) True Negative Rate (TNR = TN / (TN + FP)): to evaluate effectiveness of 'adding' the target attribute; and (iii) Accuracy. Note that in all these cases, low scores imply effective inversions.

Evaluating Attribute Obfuscation. We evaluate the uncertainty performance by comparing the posterior probabilities (using a held-out classifier F) before ($y = F(x)$) and after ($\bar{y} = F(\bar{x})$) image obfuscation. Specifically, we consider Shannon entropy $H(y)$ to measure the uncertainty (maximum entropy = 1) and additionally observe the confidence of the prediction y (maximum uncertainty at $p(y_i)$ = 0.5) to evaluate attribute obfuscation.

4.3. Evaluation against Unseen Adversary's Attack

We verify that our approach to invert attribute presence in images can better conceal inferences over sensitive attributes and generalize to an unseen adversary as compared to typical GAN-based models. In particular, we consider two baselines: IcGAN [13] trains an encoder to map images to input space of a pretrained conditional GAN. By modifying condition vectors encoded from images, IcGAN can manipulate attributes of corresponding inputs. On the other hand, StarGAN [4] combines conditional GANs with cycle consistency loss to ensure image contents. Note that attributes classifiers in both methods are designed to solely fit the real images. To evaluate robustness against unseen adversaries, we first sanitize images in the testing set for 10 attributes, respectively, and then obtain the prediction accuracy from the held-out ResNet-18. Note that the best case is to minimize accuracy because the original attributes are completely removed.

Accuracy (%)					
	Blonde Hair	Eyeglass	Heavy Makeup	Male	No Beard
D_{attr}	20.09	13.11	60.68	45.23	35.83
D_{attr}+AT	39	46.5	37.07	36.54	47.19
Ours	**10.16**	**3.61**	**14.5**	**3.79**	**13.07**
	Pale Skin	Smiling	Wavy Hair	Wearing lipstick	Young
D_{attr}	52.03	4.39	28.61	78.22	36.68
D_{attr}+AT	66.99	35.73	57.29	38.83	61.79
Ours	**35.72**	**4.18**	**28.56**	**11.51**	**29.67**

Table 2. Ablation study on three models with distinct classifiers.

Quantitative Results. Table 1 presents the accuracy after sanitization in detail. *Real* denotes the test accuracy of the hold-out classifier on test data. This indicates the generalizability of the classifier to unseen data and ensures the credibility for the following measurement. We observe that the proposed framework reaches the lowest TPR, TNR, and accuracy consistently on most attributes. We find the the prior adversarial manipulation methods IcGAN and StarGAN under-perform as they overfit to the attribute-classifier (see Section 3.2) during training. Thus, they do not generalize well to unseen adversaries, especially for challenging attributes such as *Heavy Makeup* and *Male*.

Qualitative Results. We visualize samples generated by our method and StarGAN in Figure 4. Although both methods generate realistic images, our method can better conceal attributes than StarGAN. For instance, our method completely removes pale skin (Fig. 4f) while StarGAN only focuses on some specific regions. In addition, our method adds more wrinkles to conceal the age information. In contrast, StarGAN only changes the hair color slightly. Lastly, StarGAN tends to include similar patterns to images as shown in Figure 4(c,d), while our method can provide diverse patterns over different attributes. From the qualitative results, we find promising results of our approach inverting attributes in images, while being reasonably faithful to the original input image.

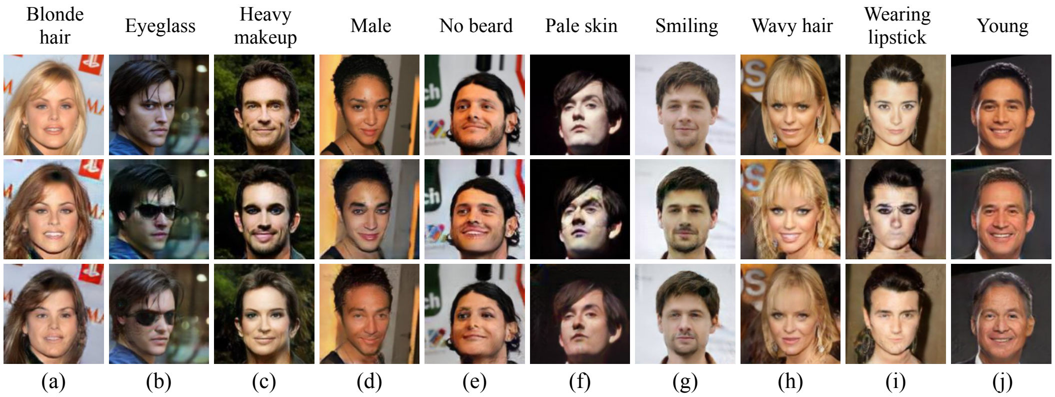


Figure 4. Attribute inversion qualitative results. (top) input images; (middle) attribute-inverted images by StarGAN; (bottom) our method.

	$\delta_2 = 0$	$\delta_2 = 0.1$	$\delta_2 = 0.2$
Privacy	0.155	0.147	0.130
Utility	0.863	0.807	0.781

Table 3. Trade-off between privacy (lower is better) and utility (higher is better).

4.4. Ablation Study

We conduct an ablation study on three models with distinct attribute classifiers to confirm the strength of the proposed bi-directional classifier as introduced in Section 3.2. In particular, (i) D_{attr} is equipped with an attribute discriminator solely updated with real data, which is in spirit of traditional image editing methods; (ii) D_{attr}+*AT* additionally performs adversarial training (AT) by additionally updating D_{attr} using generated data; and (iii) *Ours* represents the model equipped with the proposed bi-directional classifier. Note that the same encoder-decoder architecture is adopted for all three models.

The accuracy comparison among three models is reported in Table 2. We first observe that D_{attr} does not remove attributes thoroughly. Ideally, models with adversarially trained classifiers should perform better since it iteratively learns to identify private patterns. However, D_{attr}+*AT* performs worse than D_{attr}, which confirms the partial attribute inversion problem. Lastly, *Ours* reaches the best performance across all attributes. The reason is two-fold. First, updating the discriminators with generated images makes the model generalize well to unseen classifiers. Second, the proposed bi-directional classifier further prevents the confusion problem.

4.5. Analysis on Trade-off Parameters

We present a study of how distinct δ_i values balance privacy and utility. As discussed in Section 3.3, the proposed framework incorporates three parameters δ_i to control the trade-off. In practice, δ_3 is set to zero as it is related to reconstruction, and δ_1 is often set to be a small number (e.g. 0.05 for L1 norm) since we expect lower distortion. Thus, in this study we mainly focus on the changes of target and non-target attributes when different δ_2 is provided. We denote the accuracy for target attributes by *privacy* and the one for non-target attributes by *utility*.

Ideally, we want to achieve the lowest privacy leakage while maximizing utility. However, privacy and utility may not be fully independent, leading to a trade-off. As shown in Table 3, our model can adjust privacy leakage level by using different δ_2. As desired, if we allow more utility distortion (i.e. larger δ_2), the lower privacy leakage is reached, while the utility distortion is also increased. Users can find suitable parameters based on their applications.

4.6. Evaluation on Image Quality

To show that the proposed algorithm can sanitize images without significantly sacrificing image quality, we measure Fréchet Inception Distance (FID) on CelebA for both our algorithm and StarGAN [4]. We first use each model to generate 50000 images by randomly choosing one target attribute and compute the scores separately on two sets. According to the experiment, our method achieves 9.52 while StarGAN achieves 12.52, which is comparable. This justifies that our method can generate sufficiently high quality images while removing sensitive attributes.

4.7. Evaluation on Uncertainty

In the following, we show that the proposed two-stage method can secure privacy information by introducing uncertainty over certain sensitive attributes. Specifically, we consider prediction probability and Shannon entropy to measure privacy leakage. The goal of our method is to minimize leakage, which means ideally, the obfuscated images should have prediction probability 50% and 1 bit (base 2) entropy over target attributes. Moreover, since vanilla classifiers often suffer from over-confidence problems (i.e. it only outputs either 0% or 100%) and thus cause distorted evaluation, we re-train the ResNet-18 classifier with mix-up strategy [16], which mixes two input data and their labels

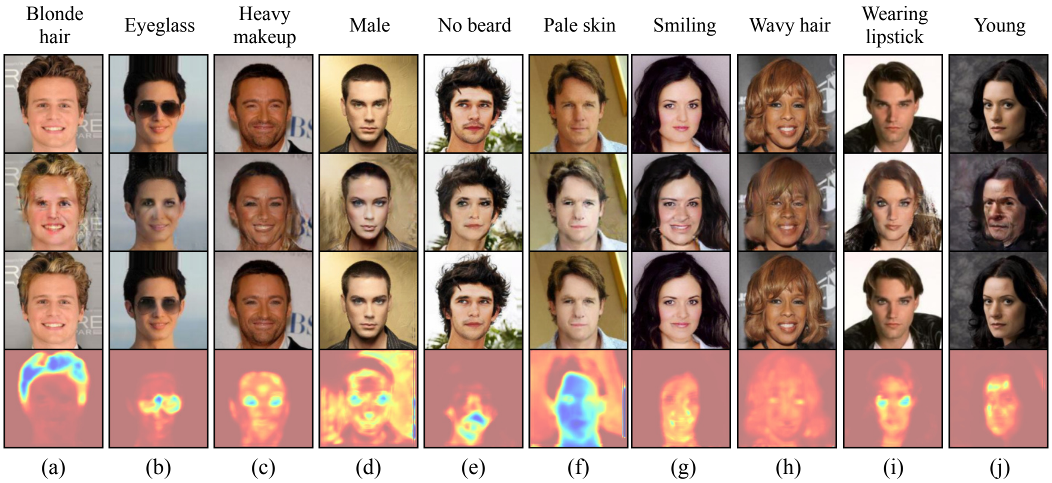


Figure 5. Visualization of obfuscated images. From top to bottom, each row presents original images, attribute-inverted images, obfuscated images, and attention maps of λ. For every column, we choose one attribute as a target.

	Blonde hair	Eye-glasses	Heavy makeup	Male	No beard	Wavy hair	Wearing lipstick	Young
				Entropy (bits)				
				Positive → Uncertain				
Real	0.56	0.59	0.53	0.12	0.13	0.57	0.43	0.24
Ours	0.75	0.82	0.77	0.84	0.87	0.66	0.82	0.79
Gain	0.18	0.23	0.24	0.72	0.74	0.09	0.38	0.54
				Negative → Uncertain				
Real	0.06	0.04	0.08	0.23	0.47	0.16	0.08	0.46
Ours	0.83	0.89	0.80	0.86	0.66	0.72	0.81	0.64
Gain	0.78	0.84	0.72	0.63	0.19	0.56	0.73	0.18
				Probability (%)				
				Positive → Uncertain				
Real	86.2	85.3	87.5	98.1	97.7	86.3	90.4	95.6
Ours	72.5	64.1	70.9	62.0	61.0	80.0	64.7	63.0
Gain	13.7	21.1	16.5	36.0	36.7	6.3	25.7	32.6
				Negative → Uncertain				
Real	0.8	0.6	1.4	4.5	10.6	2.6	1.4	10.4
Ours	44.4	47.1	37.6	38.8	22.4	38.1	38.4	20.8
Gain	43.6	46.5	36.2	34.3	11.9	35.6	37.0	10.4

Table 4. Quantitative evaluation of attribute obfuscation. Better performance at this task is indicated by higher entropy (maximum = 1 bit) and probability scores approaching 50% (i.e., chance-level). 'Real' denotes performance of adversary on original non-obfuscated images, and 'ours' on obfuscated counterparts. 'Gain' denotes the difference between the two. We evaluate on both positive (input images containing the target attribute) and negative (not containing it).

to augment the training data. With the regularization, the model allows ambiguity occurring in predictions and thus prevent over-confident predictions.

In Table 4, we report the entropy and prediction probability, for images before ('Real') and after ('Ours') obfuscation, and their corresponding difference ('Gain'). We additionally group the results into '(Positive/Negative) → Uncertain', where Positive indicates an attribute is present in the input image, and Negative indicating the attribute is absent. We observe that both entropy and probability are driven toward uncertainty by a large margin, which strongly supports the capability of the proposed two-stage method. Interestingly, we find that the Negative → Uncertain translation performs better than Positive → Uncertain most of the time. This suggests that adding new features to an image is easier than remove information from an image.

We show in Figure 5 that our model can obfuscate sensitive attributes by merging characteristics of original and attribute-inverted images although the images do not necessarily exist in the training data. To name a few, for hair color (Fig. 5 (a)), the model learns to blend blonde into black hair; for male (Fig. 5 (d)), it learns to put on light make-up on the man's face; for pale skin (Fig. 5 (f)), it learns to fuse the face colors. We additionally present interpolation pixel coefficients λ in Figure 5. Surprisingly, the model automatically identifies the regions related to sensitive attributes even though only image-level labels are provided.

5. Conclusion

In this paper, we were motivated by providing fine-grained control over private information leakage in images. Towards this goal, we presented an approach to obfuscate images, where the information w.r.t target privacy attributes is manipulated – either by inverting the attribute, or maximizing uncertainty over it. In spite of numerous challenges this setting presents (e.g. generating out-of-domain obfuscated data, generalizing to unseen attribute inference attacks), we show that images can be sufficiently altered to either introduce false information, or minimize the information content of an attribute, while maintaining the overall appearance of the original input image.

References

[1] Jianmin Bao, Dong Chen, Fang Wen, Houqiang Li, and Gang Hua. Towards open-set identity preserving face synthesis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6713–6722, 2018.

[2] Martin Bertran, Natalia Martinez, Afroditi Papadaki, Qiang Qiu, Miguel Rodrigues, Galen Reeves, and Guillermo Sapiro. Adversarially learned representations for information obfuscation and inference. In *ICML*, 2019.

[3] Tian Qi Chen, Xuechen Li, Roger B Grosse, and David K Duvenaud. Isolating sources of disentanglement in variational autoencoders. In *Advances in Neural Information Processing Systems*, pages 2610–2620, 2018.

[4] Yunjey Choi, Minje Choi, Munyoung Kim, Jung-Woo Ha, Sunghun Kim, and Jaegul Choo. Stargan: Unified generative adversarial networks for multi-domain image-to-image translation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8789–8797, 2018.

[5] Elliot Creager, David Madras, Jörn-Henrik Jacobsen, Marissa A Weis, Kevin Swersky, Toniann Pitassi, and Richard Zemel. Flexibly fair representation learning by disentanglement. In *ICML*, 2019.

[6] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in Neural Information Processing Systems (NIPS)*, pages 2672–2680, 2014.

[7] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016.

[8] Zhenliang He, Wangmeng Zuo, Meina Kan, Shiguang Shan, and Xilin Chen. Attgan: Facial attribute editing by only changing what you want. *IEEE Transactions on Image Processing (TIP)*, 28(11):5464–5478, 2019.

[9] Mehdi Mirza and Simon Osindero. Conditional generative adversarial nets. *arXiv preprint arXiv:1411.1784*, 2014.

[10] Augustus Odena, Christopher Olah, and Jonathon Shlens. Conditional image synthesis with auxiliary classifier gans. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 2642–2651, 2017.

[11] Tribhuvanesh Orekondy, Mario Fritz, and Bernt Schiele. Connecting pixels to privacy and utility: Automatic redaction of private information in images. In *CVPR*, 2018.

[12] Tribhuvanesh Orekondy, Bernt Schiele, and Mario Fritz. Towards a visual privacy advisor: Understanding and predicting privacy risks in images. In *ICCV*, 2017.

[13] Guim Perarnau, Joost Van De Weijer, Bogdan Raducanu, and Jose M Álvarez. Invertible conditional gans for image editing. *arXiv preprint arXiv:1611.06355*, 2016.

[14] Andrew Perrin and Monica Anderson. Share of us adults using social media, including facebook, is mostly unchanged since 2018. *Pew Research Center*, 10, 2019.

[15] Proteek Chandan Roy and Vishnu Naresh Boddeti. Mitigating information leakage in image representations: A maximum entropy approach. In *CVPR*, 2019.

[16] Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. *arXiv preprint arXiv:1710.09412*, 2017.