StyleID: Identity Disentanglement for Anonymizing Faces

Minh-Ha Le
Linköping University
Sweden

Niklas Carlsson
Linköping University
Sweden

ABSTRACT

Privacy of machine learning models is one of the remaining challenges that hinder the broad adoption of Artificial Intelligent (AI). This paper considers this problem in the context of image datasets containing faces. Anonymization of such datasets is becoming increasingly important due to their central role in the training of autonomous cars, for example, and the vast amount of data generated by surveillance systems. While most prior work de-identifies facial images by modifying identity features in pixel space, we instead project the image onto the latent space of a Generative Adversarial Network (GAN) model, find the features that provide the biggest identity disentanglement, and then manipulate these features in latent space, pixel space, or both. The main contribution of the paper is the design of a feature-preserving anonymization framework, StyleID, which protects the individuals' identity, while preserving as many characteristics of the original faces in the image dataset as possible. As part of the contribution, we present a novel disentanglement metric, three complementing disentanglement methods, and new insights into identity disentanglement. StyleID provides tunable privacy, has low computational complexity, and is shown to outperform current state-of-the-art solutions.

KEYWORDS

Identity disentanglement, anonymization, feature-preserving, privacy, StyleGAN, face editing

1 INTRODUCTION

Machine learning is currently considered one of the primary enablers for future technological advancements. AI technologies operating in environments including humans are therefore expected to need access to large datasets containing images of humans.

However, large image datasets containing real faces quickly cause privacy concerns. For example, the police's use of ClearView AI to track criminals over the world has received significant media spotlight [11]. With 3 billion images of people in the database, concerns have been raised regarding who is in the database and the police's right to use it for surveillance. The privacy risks of such image databases become even greater if considering the potential consequences of others using the databases for their own benefit.

Here, it is important to note that it typically is not the AI technologies themselves that present the privacy risks; it is the datasets, and how the datasets and the models created using these datasets are being used. To prevent misuse of the datasets while at the same time enabling the development of future applications in a privacy-conscious manner, it is therefore important that datasets can be anonymized in ways that preserve the utility of the datasets. If done well, such tools (and the generated datasets) will benefit machine learning technologies (that require training using datasets containing images of humans) and other fields (e.g., image editing, synthetic 2D/3D-avatars, privacy on social media).

Anonymization can easily be done using occlusion and confusion methods that hide the identity of the faces. However, such methods typically significantly reduce the utility of the datasets and the accuracy that can be expected by the machine learning models trained on such data. Anonymizing facial datasets in ways that preserve the facial characteristics of both the individual images and the dataset as a whole is a much harder task. One reason for this is that facial images represent one of the most complex information types and faces provide a direct identity representation of humans. This is perhaps why most prior work primarily has focused on the naturalness of the anonymized faces [7, 15, 18, 25] or proved basic properties such as k-anonymity [8, 28].

In this paper, we take a more ambitious approach in which we disentangle and hide the identity-related facial features while aiming to preserve the main visual characteristics of the faces. To achieve this objective, we present novel identity disentanglement approaches that operate in latent space, pixel space, or both. The solutions presented are part of our framework, called StyleID, that (1) identifies and manipulates the identity-relevant information in a face to provide an anonymized face, while (2) preserving non-identity-related features (e.g., pose, facial expression, background, and hair), (3) without destroying the facial naturalness. As of today, several papers have shown how GANs can be used for one or two of these aspects at a time. (See Sec. 8.) However, to our knowledge, we are the first to address all three aspects simultaneously.

Much of the de-identification works do not preserve attributes and/or lack in naturalness [7, 21, 42]. Other works use pre-trained image generators such as StyleGAN [16, 17] to achieve high naturalness, but often only focus on the manipulation of attributes in the facial images [9, 31, 36, 43]; not de-identification. Overall, there is very limited work studying the privacy-sensitive information in both latent space and pixel space.

Contributions: The main contribution of this work is the design of a *feature-preserving anonymization framework* that uses our new approach to protect the individuals' identity, while preserving as many characteristics of the original faces in the image dataset as possible. However, in the derivation of this design, the paper makes several additional important contributions. First, at the core of the design are three novel methods for *identity disentanglement*. The methods are complementing each other, build upon each other, and operate either in latent space (only) or in both latent and pixel space simultaneously. For example, the first method identifies and manipulates the part of the latent space that provides the most

Proceedings on Privacy Enhancing Technologies 2023(1), 1–15

https://doi.org/XXXXXXX.XXXXXXX(TBA)

disentanglement, the second method uses segmentation masks together with insights from the first method to generate random faces for which we control the overlap in pixel space, and the third method builds a model that automates the disentanglement (and anonymization) process that operates in latent space but is trained using a ground truth built upon the second method.

Second, we present a novel identity disentanglement metric that we use to derive insights, evaluate methods, and demonstrate that effective identity disentanglement is possible in latent space. The results of our evaluations provide new insights into how to best hide privacy-sensitive information in both latent and pixel space.

Third, we use these insights to incorporate the methods into our StylelID framework so as to provide tunable anonymity and attribute preservation tradeoffs. The framework allows us to transform the identity-relevant information in a face into an anonymized face with a desirable level of anonymity, while preserving identity-irrelevant features and the naturalness. Furthermore, the methods are efficient (e.g., the first two methods only use pre-trained models) and outperform current state-of-the-art anonymization solutions.

Outline: Sec. 2 presents an overview of the framework, our disentanglement methods, and defines our disentanglement metric. Secs. 3 and 4 present how anonymization can be achieved by manipulating layers or channels of the latent codes associated with a face. Our feature-aware identity masking method (Sec. 5) and the latent swapper (Sec. 6) are presented next. Sec. 7 evaluates StyleID (and its methods) against facial recognition tools, based on how well they preserve attributes and the identity diversity they provide. Finally, Sec. 8 compares with related works, Sec. 9 discusses security and ethical considerations, and Sec. 10 presents our conclusions.

2 FRAMEWORK OVERVIEW

This paper focuses on the *anonymization* of image datasets including faces. In addition to removing identifying characteristics so to protect the identity of individuals (i.e., de-identification) and ensuring that the faces are altered in such a way that each face no longer can be related back to a given individual (i.e., the process should not be revertible), a good anonymization technique should ensure high utility of the resulting data.

To ensure high utility, we present the design of StyleID, a *feature-preserving anonymization framework* that transforms the identity-relevant information in a face into an anonymized face of the desirable level of anonymity, while preserving as many non-identity-related features (e.g., pose, facial expression, image background, and hair) as possible and maintaining the face's naturalness.

At the core of the design is the basic idea of first applying identity disentanglement in latent space and then applying changes to the aspects of a face (in latent space, pixel space, or both) that provide the most attractive tradeoffs between anonymization and feature preservation. For much of this manipulation we leverage the latent space of StyleGAN [16, 39]. In addition to allowing facial editing in latent space, StyleGAN has been found (by human evaluators) to produce more realistic/natural images than other state-of-the-art solutions [44]. With the naturalness of the generated images of any framework being bounded by the quality of the face generator used, our choice to use StyleGAN's generator allows us to achieve higher naturalness than most prior anonymity frameworks [15, 25, 28, 35].

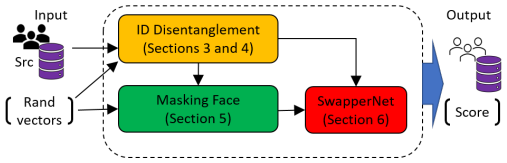


Figure 1: Overview of our StyleID framework.

2.1 Identity disentanglement approaches

We present three complementing disentanglement approaches and show how to use them to achieve our anonymization objectives.

Disentanglement in latent space (Secs. 3 and 4): The first approach operates in latent space. With this approach, we (1) project the face images into latent space, (2) identify the layers/channels in the latent code that contribute the most to the individual's identity, (3) manipulate these layers/channels of the latent code in a desirable direction (e.g., as far away from the original source face or towards an alternative target identity), before (4) generating the final image. For tunable privacy, we control how far away from the source identity we push the identity. While this approach performs well, it is not able to (on its own) consistently preserve the background and some facial features (e.g., hair, facial expression, eyes direction) and may create problems when applied to video.

Disentanglement in pixel space (Sec. 5): We then present a technique that incorporates the use of *segmentation masks* to improve the quality of the generated images and to better preserve selected features as seen in pixel space. This approach is attractive for videos and other contexts that place stricter requirements on preserving specific facial features and/or the background. Using an example implementation, we demonstrate how the use of segmentation masks combined with some of the insights from our first approach can be used to generate random faces that have the same face mask and how this can significantly simplify accurate face swapping. The approach allows us to effectively generate faces with a matching pose, provides fine-grained control of how much of the original identity (and face) is preserved in the generated output image and is relatively lightweight. While this approach solves some of the problems of the first approach, we have found that it can result in a mismatch of lighting between the randomly generated face and the original face. Although such issues in most (but not all) cases can be nicely corrected by a match color process, we note that such a process requires additional care.

Latent swapper (Sec. 6): Finally, to address the remaining shortcomings, we build a model that automates the process of anonymizing a face in latent space with a ground truth built using our segmentation mask approach. The model finds an α-mask and trains swapper modules weighted based on the insights derived from our latent space approach to hide the source identity beyond the identification threshold of modern facial recognition systems (FRS:s) while limiting the changes to non-identity related attributes.

Fig. 1 presents an overview of StyleID and the three disentanglement methods designed and implemented within the framework. (1) The *ID Disentanglement* component (yellow) is at the heart of our design. It is used to disentangle identity information in latent space. (2) *Masking Face* (green) is applied in pixel space. This component is used to generate and apply random identity masks. (3) *SwapperNet* is an all-in-one anonymizer model that swapps the original identity

to a random identity. While each of the three components builds on the prior component(s), each component can produce anonymized faces also without the use of the later component(s).

2.2 Privacy-utility tradeoff

The main problem of anonymizing datasets is finding a desirable tradeoff between privacy and utility. One challenge is the lack of universal metrics to quantify the tradeoff. Here, we present and motivate the primary privacy and utility metrics used in this work.

First, motivated by the importance of protecting the facial identity against FRS:s, we use the identity distance calculated using popular face embedding models used by such systems to measure privacy. Ideally, an anonymized face o should have an identity distance to the original source face that exceeds some threshold or is considered "far-enough" away from the source. Second, we measure the utility using attribute scores extracted from both the source faces and the generated output faces. We next formalize the above metrics of privacy and utility, as calculated over a full dataset.

Definition (Privacy metrics): Let $\mathcal{A}$ be an anonymizer converting a source dataset $\mathbb{S}$ containing N human faces $\{S_1, S_2, ..., S_N\}$ into an anonymized output dataset $\mathbb{O}$ containing N output faces $\{O_1, O_2, ..., O_N\}$. Furthermore, let $IdNet$ be a face embedding model. Now, the identity distances between a given source face S_i and the corresponding output face O_i can be calculated as follows:

$$\Delta_i^{ID} = \delta(IdNet(S_i), IdNet(O_i)), \tag{1}$$

where $\delta()$ is the pairwise distance calculated over the embedding vectors $e_i^S = IdNet(S_i)$ and $e_i^O = IdNet(O_i)$, for $1 \leq i \leq N$. Using this distance, we define the privacy metric I as the average identity distances of all the source-output pairs; i.e., $I = \frac{1}{N}\sum_{i=1}^{N} \Delta_i^{ID}$.

For the purpose of evaluation, we also report the probability that an arbitrary pair satisfies a privacy threshold Γ and consider the anonymizer $\mathcal{A}$ to "strictly" satisfying a privacy guarantee Γ if $\Delta_i > \Gamma, \forall O_i \in \mathbb{O}$). We also calculate the ROC curve and report the average rank of the output image O_i when comparing the output images distances to the source image S_i.

The above metric and statistics are based on the decision process of current state-of-the-art FRS:s, not on probability theory. Future work could involve the use of a more formal privacy definition.

Definition (Utility metrics): Again, consider source set $\mathbb{S}$, an anonymizer $\mathcal{A}$, and a corresponding output set $\mathbb{O}$. Furthermore, let $AttrNet$ be a face attribute classifier model that extracts M attributes $\{a_1, a_2, ..., a_M\}$ from the face, where $a_i \in (0, 1)$, $1 \leq i \leq M$. Now, the attribute distance between a source-output pair i $(1 \leq i \leq N)$ can be calculated as:

$$\Delta_i^{Attr} = \delta(AttrNet(S_i), AttrNet(O_i)), \tag{2}$$

where $\delta()$ is the pairwise distance calculated over the attribute vectors $a_i^S = AttrNet(S_i)$ and $a_i^O = AttrNet(O_i)$. Using this distance, we define the utility metric A as the average attribute distances of all corresponding pair source-output faces; i.e., $A = \frac{1}{N}\sum_{i=1}^{N} \Delta_i^{Attr}$.

For evaluating utility, we also measure the anonymizer's ability to preserve the distribution of the attributes observed in the dataset and the identity diversity.

2.3 Identity disentanglement metric

Identity disentanglement in latent space has not been addressed by prior work and is not possible with existing methods [9, 31, 43]. One main challenge is that the identity is a complex combination of several facial features. To perform identity disentanglement, we first define a metric to measure the identity disentanglement achieved by an anonymizer $\mathcal{A}$. Ideally, the metric should be sensitive to identity changes and insensitive to non-identity-related changes.

To achieve this goal, we try to simultaneously maximize the identity distance and minimize the attribute distance. For this reason, our metric combines the privacy and utility metrics from the previous subsection. In particular, we calculate the disentanglement score going from face S_i to O_i as follows:

$$IA_{score}(S_i, O_i) = \alpha \cdot h(\Delta_i^{ID}) - \beta \cdot h(\Delta_i^{Attr}), \tag{3}$$

where α and β are tunable constants, and $h()$ is a normalization function (max-min scaling) that normalizes identity and attribute distances into the same value range. Through careful selection of α and β we have found the metric to nicely capture how successful a change from face S_i to O_i was at modifying the identity without significantly changing the facial attributes. For the experiments, we use α=1, β=1.25, and the Euclidean pairwise distance δ.

Implementation details: We use several pre-trained models in our experiments. The image generator G used is a StyleGAN2 model [39] trained on the FFHQ dataset [16] at 1024x1024 resolution. We use pSp [33] (trained on the same dataset) as our image encoder. We use the state-of-the-art facial recognition models CurricularFace [14] and ArcFace [3] to calculate the identity embeddings of $IdNet$ (used to calculate identity distances). Finally, the attribute predictor $AttrNet$ used is a custom MobileNet model [12] that is trained on the CelebA dataset [23] to predict 40 binary attributes. The predictor outputs a confidence vector in the value range (0,1).

2.4 Tunable anonymity

All our disentanglement methods are designed to allow tunable anonymity (define next) for the levels of privacy targeted.

Definition (Tunable anonymity): *Given a source face $\mathbb{S}$, an anonymizer $\mathcal{A}$ should be able to adjust the level of privacy offered based on the parameters assigned for the specific use-case scenario.*

Here, we consider and target three levels of privacy:

- *Low:* Small modification to identity features $I \approx \Gamma$, where Γ is the detection threshold of the FRS, that are sufficient to fool FRS but that are barely noticeable to the human eye (e.g., privacy filters that appear like noise to a user). In this case, ideally all M attributes from the source $\{a_0, a_1, ..., a_M\}$ are preserved, meaning that A is greater than a threshold Θ or $AttrNet$ successfully extracts all attributes $a_i > \theta, \forall O \in \mathbb{O}$, where θ is the binary decision threshold of $AttrNet$.
- *Medium:* Preserving a subset of facial features $\mathbf{Q} \in \mathbf{M}$ that result in identity changes both to an FRS $I \approx \Gamma$ and humans, i.e., $\forall O \in \mathbb{O}$, there exists $a_i > \theta, \forall a_i \in \mathbf{Q}$.
- *High:* One may also want to enforce desirable attribute distribution properties among the set of generated faces (e.g., t-closeness and l-diversity [20]). This is to limit the individual-specific information revealed by the observed face attributes.

For each of the three proposed disentanglement approaches, we demonstrates how to tune the level of anonymization among the first two levels ("low" and "medium"). To achieve the highest level of anonymization, one can add an outer loop on-top of our framework so as to ensure that some desirable attribute-distribution properties (e.g., l-diversity, t-closeness) are satisfied. However, such extensions are considered outside the scope of this paper. Here, we just note that our ability to control selected attributes in the generated images enables also this level of anonymization (if desired).

3 IDENTITY DISENTANGLEMENT AND SWAPPING LAYERS IN LATENT SPACE

3.1 Background: StyleGAN + latent space

StyleGAN is a generative model capable of generating realistic images from random noise vectors. With StyleGAN, the input latent codes $z \in Z$ are passed through a mapper, which is a sequence of fully connected layers outputting intermediate latent codes $L \in W$. Here, L is a two-dimensional array of size 18×512, where rows are called layers and the values in each layer are called channels.

One attractive property of StyleGAN is that its intermediate latent space W is highly disentangled. For example, Karras et al. [16] observed that certain layers in W correspond to specific subsets of facial features and that the layers can be split into three categories: *Course layers* (0 to 3) represent high-level attributes such as pose, face shape, hairstyle, and eyeglasses. *Middle layers* (4 to 8) represent features such as the structure of the eyes, mouth, and nose. *Fine layers* (8 to 17) hold the color scheme and micro-structure.

Prior work has shown how such disentanglement can be leveraged to manipulate selected facial features [9, 31, 36, 43]. By carefully manipulating the latent codes these works have successfully changed one feature at the time without changing the identity of the face. However, thus far no work has considered the opposite problem; i.e., how to change the identity while preserving other facial features. In this section, we consider such disentanglement.

3.2 High-level approach and key problems

We next outline the four key steps of our approach for disentanglement in latent space. Details are provided in later subsections.

Steps 1+2 (Identity disentanglement): We first identify the channels/layers in latent space providing the most desirable identity disentanglement. We call this the *identity disentanglement problem.*

Problem definition (Identity disentanglement): *Given an input face S and a GAN model G with latent space W, assuming that we have a projector $\mathcal{P} : S \rightarrow W$ to project face $S \rightarrow L_S$, the problem consists of identifying the layers/channels in the latent codes L_S that provide the most identity disentanglement according to our disentanglement metric (equation (3)) that measures both the change in the identity and the attributes of S when manipulating L_S.*

To solve this problem, the main idea is to find an identify direction that turns the identity of a face image away in such a way that other facial attributes are not affected. Ideally, when the identified layers (or channels) are changed, only the identity should change while other (non-identity related) attributes are preserved.

Note that prior works that manipulate one feature at a time in latent space [1, 9, 31, 36, 40, 43] are not applicable in our context. One reason is that those attributes are easy to classify into binary classes or can easily be mapped to a scale between 0 and 1. In contrast, with identity disentanglement there is no visible scale onto which all face images easily can be mapped. To identify the layers to manipulate, we instead use our identity disentanglement metric to evaluate the disentanglement achieved when anonymizing a large set of images as per steps 3+4 (but without knowing the best layers to select). However, before we present how this is done, we first present a brief introduction to how steps 3+4 are done when the best layers (or channels) already have been identified in step 2.

Steps 3+4 (Feature-preserving anonymization): We next manipulate the identified layers (or channels) of the latent code L_S away from the original identity or towards an alternative identity T that has been randomly generated. Finally, we use an encoder G to generate an output image O using the modified latent code $L_{S'}$. We next formalize the problem these two steps aim to optimize.

Problem definition (Image anonymization using latent code): *Given a set of source images $\mathbb{S}$, the problem is to anonymize the images so that each of the anonymized images $O_i \in \mathbb{O}$ have maximum identity disentanglement distance from their respective source image S_i given some desirable level of anonymity.*

Practically, this means that we would like to push the identity distance (weighted by α in equation (3)) as far away from the original identity while keeping the attributes (weighted by β) as close to the original identity as possible, given a desirable level of anonymity. Here, the selected layers, the selected target face T (randomly generated), as well as the distance that each code is pushed all impact the level of anonymity (and identity disentanglement) achieved. Fig. 2 shows a high-level overview of the disentanglement steps taken to modify the latent code L_S of an image S using a subset of layers/channels of the latent code L_T of a target identity T, and the evaluation of a resulting output image O.

3.3 Mapping to latent space (step 1)

For the work presented here, we use a variant of StyleGAN called pSp [33] that uses an enhanced latent space $W+$. One significant difference of pSp is that it has the capability to encode a real image to latent space. After such a mapping has been done, the disentangled latent code can easily be used for image editing of different kinds. In pSp, the layers of W codes are passed through a number of small fully connected convolutional networks to achieve the extended latent code in $W+$, which is proved to encode more information than in W. To simplify the notation, throughout the reminder of paper, we use the term latent space W to refer to both W and $W+$, and unless explicitly stated use the latent codes provided by pSp.

3.4 Finding layers to modify (step 2)

Our method for finding the best subset of layers (or channels) to modify in latent space is to evaluate the disentanglement that a large number of random pairs of faces S and T achieve when swapping different subsets of their respective latent codes L_S and L_T.

The evaluation of one such test is illustrated in Fig 2 and the results are presented in Sec. 3.7. In the figure, L_S and L_T are the source and target latent codes corresponding to the source S and target identity T (for the purpose of anonymization the target identity is randomly generated by a face generator G). The gray area of the latent code $L_{S'}$ illustrates the swapped layers. To estimate

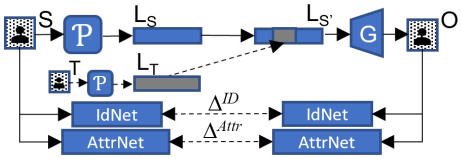
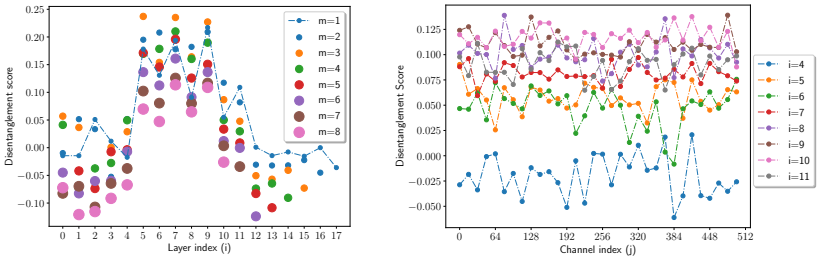


Figure 2: Identity disentanglement approach in latent space.



(a) Scoring for different subsets of layers **(b) Scoring for each channel**

Figure 3: Disentanglement scores of swapping layers and channels

the identity disentanglement score we take the result $L_{S'}$ (equal to L_S where the identity is swapped to the random identity in L_T), and pass it through the face generator G. The corresponding generated faces from L_S and $L_{S'}$ are then be passed through $IdNet$ to calculate the identity distance and $AttrNet$ to calculate attribute distance. Finally, those distances are used to calculate the identity disentanglement scores. By symmetry, to evaluate the impact of swapping layers, we do the same swapping and evaluation of the impact this has on T. For clarity, this is omitted from the figure.

To keep this analysis feasible (there is exponentially many combinations of layers and evaluation is time consuming), we used a window-based approach in which we swap groups of m consecutive layers; i.e., all layers from layer i through layer $i+m-1$. The use of consecutive layers is further motivated by having found that neighboring layers often contribute to similar (or the same) features. We then select the identity direction as the (i, m) values that maximizes the average identity disentanglement score (calculated using equation (3) between a large set of source images S and the generated output images $O = G(L_{S'})$ obtained when swapping layers $\mathcal{L}_{i,m} = (i, i+1, ..., i+m-1)$ of latent code L_S with the same layers of the latent code L_T of some random image T.

3.5 Manipulating the identity (step 3)

We consider two ways of manipulating a selected set of layers $\mathcal{L}$ in the latent code: (1) by swapping layers $\mathcal{L}$ of the source identity S with the same set of layers of a target identity T, and (2) by pushing the source identity S in some direction to obtain a new identity S'.

For the first case, we use the same swapping methods as discussed in the previous subsection and illustrated in Fig. 2 but this time with S and T as the two input images. For the second case, we simply pick the identity S' that maximizes the identity disentanglement score from a large set of randomly generated images where we manipulate a selected set of layers in the latent code. Here, we select to modify the set of layers that we found provided the best results over a large number of example images.

3.6 Dataset and training-evaluation split

Unless explicitly stated, we use the CelebAMask-HQ dataset [19] dataset in our experiments. The main reason for this choice is that each face image in the dataset is annotated with identity, 40 binary attributes, and it comes with segmentation masks that we use for our analysis in Sec. 5. In total, the dataset includes 30,000 images of 6,217 identities. In addition, the annotated attribute values provide ground truth for the training of our $AttrNet$ model. Finally, we have found that the dataset's high resolution is valuable for ensuring low identity loss when projecting a face onto latent space W.

For our evaluation, we split the dataset into three non-overlapping sets of identities: the source set, the target set, and a validation set (used only for validation). The first two sets (source + target) each include 2,000 unique identities (≈10,000 images). The validation set contains the remaining 2,217 identities. For simplicity, each identity is represented using (only) a single random sample image.

3.7 Results when swapping layers

We have found that modifying 2-to-4 layers often produce the best results. Comparing the individual starting points i, we have found that layers 5-9 typically provide the best results, with layers 5, 7, and 9 (individually) contributing the most to the disentanglement. This is shown in Fig. 2a, where we show the disentanglement scores for different combinations of starting points i (on x-axis) and window sizes m (markers of different color and size). We use a dotted line to show the baseline when only switching a single layer (m=1). From the figure it is clear that the best choice (on average) is to use layers 5-7 (assuming we use consecutive layers). Furthermore, the distinct spikes seen for layers 5, 7 and 9 show that these layers individually provide the highest identity disentanglement.

The last observation raises the question whether greedily using layers (5,7,9), would improve the results over using the best consecutive layers (5,6,7). To answer this question, we have compared the two candidates head-to-head and present visual results next.

3.8 Example results

Fig. 4 shows representative example results (bottom two rows) generated using the latent code L_O obtained by greedily replacing the top-three individual layers (5,7,9) or the top-three consecutive layers (5,6,7) of the latent code L_S of the source face (top row) with the corresponding layers of the latent code L_T of the target face (second row). The example faces were randomly selected from the source and target sets. While the approach of swapping layers allows us to move the identity towards that of the target, we have observed some weaknesses. First, we have found that the attributes such as the age and gender often are dependent on the target face. Since this is generated randomly, a naive implementation may in some cases therefore change gender and age. We provide further analysis on the correlation of identity and individual attributes in Appendix A. To address this shortcoming, we provide two solutions: (1) a greedy approach to optimally preserve all attribute (described in the next section) and (2) the use of available semantic editing frameworks to control the attribute (discussed in Sec. 7.2). Second,

Figure 4: Example results swapping different layers of the latent codes of the source (top row) and target (second row). Results swapping layers (5,6,7) are shown on the third row and results swapping layers (5,7,9) are shown on row four.

while the background is similar for most cases, there are clear changes (e.g., in second column). To handle these cases, we suggest combining the results with pixel-space manipulations; e.g., as with the segmentation masks used in Sec. 5.

Finally, while both approaches preserve the majority of facial features of the source image, we have found that greedily switching the best individual layers (bottom row) better preserves the properties of the source images and that switching consecutive layers (third row) results in a face somewhat closer to the target identity.

4 CHANNEL MANIPULATION

In the context of non-identity related features, prior research [9, 43] has found that more fine-grained control of facial features is possible when manipulating individual channels. In this section we investigate to what degree it may (or may not) be beneficial to swap a subset of channels rather than the full layers.

4.1 Swapping individual channels

Let's first consider the impact that each channel has on the identity disentanglement score. Fig. 2b shows the results broken down for each of the (intermediate) layers that we found provided the most disentanglement. To ease visualization, we show the average over blocks of 16 channels. When comparing Figs. 2b and 2a, we make several interesting observations. First, swapping channels appears to provide more effective identity disentanglement than swapping layers. For example, while the observed disentanglement score when swapping channels (e.g., peak of 0.125 in Fig. 2b) is smaller than the observed disentanglement score when swapping 1-to-8 layers (e.g., 0.25 in Fig. 2a), the rate of change in the disentanglement score per channel is higher when considering that a single layer includes 512 channels. Second, we observe significant differences between the disentanglement achieved by swapping different channels. Third, several of the top spikes (Fig. 2b) are associated with some of the layers (Fig. 2a) that achieved the largest identity disentanglement (if switched as a whole). However, we also see some clear differences. For example, the best scores when considering consecutive layers (Fig. 2a) belongs to layers (5, 6, 7). However, swapping channels in these layers typically results in smaller disentanglement than swapping channels for the higher numbered layers 8, 9 and 10 (Fig. 2b). These observations also show that the disentanglement score is not additive.

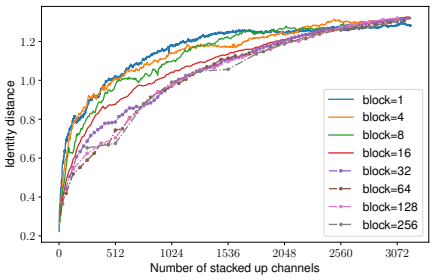


Figure 5: Block size on identity distance

4.2 Block-size evaluation

We next consider the number of channels that must be switched to achieve different levels of anonymity and the impact of grouping individual channels into blocks that either are switched or not. Fig. 5 summarizes these results. Here, we show the identity distance between the source face and the generated output face.

A few observations are noteworthy. First, there are diminishing returns with the gains of most block sizes flattening out after greedily swapping approximately 2,000 channels (on a per-block basis). Here, it should be noted that this corresponds to roughly four layers (which together have $2{,}048 = 4 \times 512$ channels). Second, the difference between using block sizes of 32-256 is small (and similar to switching entire layers, which contain 512 channels). This suggests that we either may want to use blocks smaller than 32 or we might as well swap entire layers. Third, while the smaller block sizes see the fastest improvements in identity distance, there is an infliction point around 2,600 channels (similar to 5 layers) where larger block sizes are able to achieve larger identity distance. At this point the identity distance is around 1.25 (a threshold at 99% accuracy of facial recognition model ArcFace [3] on LFW benchmark [13]). Since the larger block sizes often result in visually more appealing images, for cases where we want even greater anonymity than 1.25, it is therefore typically desirable to swap layers rather than individual channels. On the other hand, if the required level of anonymity is less, then less channels are needed to be switched if using smaller block sizes. Finally, for the case when using the anonymity threshold of 0.9 (a threshold at 95% accuracy of facial recognition model ArcFace [3]), anonymization is easily achieved with any block size, although significantly fewer swaps are needed to achieve this threshold when using smaller block sizes.

4.3 Visual example results

We now turn to the visual results. Consider first (relatively) course-grained channel swapping. Fig. 6 shows example results when swapping the 8 top-scoring blocks with size of 256. As desired, our method is able to generate an output identity (bottom row) that has been pushed away from the source identity (top row) and towards

Figure 6: Example result using coarse-grained channel swapping. Here, we swap the 8 top-scoring blocks of size 256. The three rows correspond to the source (top row), target (middle row), and the generated output image (bottom row).

the (random) target identity (second row), while preserving many of the facial properties of the source image.

While using more channels or more fine-grained channel swapping result in less control of what features are preserved, we have found that such channel swapping still can be effective in hiding the identity from facial recognition tools. For example, consider the case when we treat each channel independently (i.e., block size of 1). Fig. 7 shows example results where we swapped the 500-2,500 top channels. Notably, with less than 1,000 channels swapped, both the identity and attributes do not change significantly to the human eye, but change enough to trick facial recognition tools (ensured by the optimization done using *IdNet*). For example, the identity distance for 1,000 channels in all cases exceeds the minimum threshold of 0.9 (see Fig. 5). This demonstrates that the solution approach can be useful for somebody that wants to post images that allow their friends to recognize them but that still trick facial detection tools. When we swap more than 1,000 channels, increasingly many identity features and attributes are taken from the target images. However, important attributes such as gender, facial expressions, pose, lighting are well preserved. Even in the case when we swap 2,500 channels, we see that context features such as background, hairstyles, lighting are well preserved. Compared to swapping layers, this case could still be considered as an improvement.

On the negative side, we lose some control of which features are preserved when using fine-grained blocks. This is observed by somewhat more noticeable variations in the way that each image is anonymized. For example, while almost all attributes are the same for some images, there are a few cases with noticeable changes in the facial expression. However, in general, the method creates relatively similar images while still fooling facial detection techniques.

5 FEATURE-AWARE IDENTITY MASKING

While the solutions presented in previous sections provide an effective method for anonymizing individual images, these solutions are not sufficient on their own. To better preserve the background and to address some odd effects if applied in the video context, we propose a method that incorporates the use of segmentation masks.

High-level approach: Our key idea is to generate a face that has the same pose as the source image and then swap a selected part of this face with the randomly generated target face. To do so, we first extract the face segmentation mask of the source image, use this face mask to generate a target face that has a similar segmentation

Figure 7: Example result using fine-grained greedy channel swapping. Here, we swapped the 500-2,500 top channels of the source (top row) with a random target (second row).

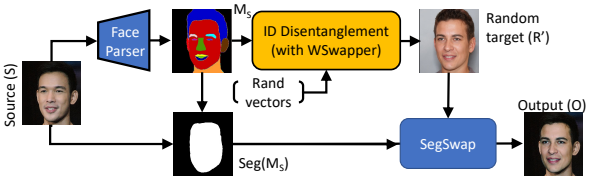


Figure 8: Feature-aware identity masking

mask, and then swap a selected part of the source face and the target face. Fig. 8 presents an overview of this approach.

Mask-based swapping: This is a non-trivial task but one for which fortunately several face-swapping models already provide excellent support. In fact, this problem can in part be well handled by face-swap models such as faceswap [2], DeepFaceLab [32], FSGAN [29]. The big difference between the existing face-swap models and our framework is that while their objective typically is to blend in a target identity, we try to hide the identity of the source face. In particular, face-swap solutions try to turn the face into that of a specific target identity, while we have less strict requirements on the final identity's relationship to the target identity. Instead we have a higher need for hiding the source identity, while still preserving facial features and producing images of high utility.

One important question that arise here is how much of the facial area needs to be changed to allow different degrees of anonymity. Fortunately, as seen in Secs. 3 and 4, most of the disentanglement were achieved by manipulating latent codes that preserved the pose and mostly affected central areas around the eyes, nose, and mouth. These areas are often well captured by segmentation masks.

Mask generation: For the generation of a precise segmentation mask, we use existing GAN segmentation models, including MaskGAN [19] and pix2pixHD [41]. These models are becoming widely used and have been shown to achieve high accuracy and performance. The use of existing models also avoids the need to train new GAN models (which can be highly time consuming).

Generation of random face: Given a segmentation mask, the next important step is to generate a random face that shares the same segmentation mask. One straightforward approach to do this is to use the segmentation mask as input to StyleGAN's encoder (e.g., pSp [33] in our case) in a similar way as with in-painting methods [25, 42]. However, this approach does not work well for our purposes since a given segmentation mask often is based on a particular face. If that face was used to train the model, the generated face is therefore likely to closely resemble the original identity. In contrast, for the purpose of anonymization, we need the newly generated face to be completely random.

To generate a random face with the same segmentation mask, we note that also a segmentation mask has a corresponding inverted latent code $L_M \in W$. Now, by exchanging a portion of this latent code that is highly correlated to the identity, we can create a rough effect of identity swapping. We have found that using this simple trick allows the segmentation mask to be used as a proxy between the original face and the newly generated face, while still ensuring that the two faces have the same segmentation mask.

Formally, given a face S and its latent code $L_S = \mathcal{P}(S)$ in latent space W, we randomly sample a vector $z \in \mathcal{Z}$ and project it onto W to get a latent code $L_R = p(z)$ of a random face $R = G(L_R)$, where $p : \mathbf{Z} \rightarrow W$ is a mapper to map Z space and W space. The segmentation mask M_S (extracted from S) is then projected onto latent space W to get the latent code $L_{M_S} = \mathcal{P}(M_S)$. The trick for generating a random face R' that has the random identity R and shares the segmentation mask M_S with S is to swap the components that have the highest identity disentanglement scores in L_R onto S; i.e., $L_{R'} = WSwapper(L_S, L_{M_S}) = L_S \cdot M_W + L_{M_S} \cdot (1 - M_W)$, where $WSwapper(.,.)$ is a swapping function that uses the identity layers/channels mask M_W in W. After that we can get random face $R' = G(L_{R'})$ sharing a segmentation mask with S. Finally, we can generate an output face O by selectively swapping the masked area $Seg(M_S)$ of choice between the source face S and the newly generated face R' (that share segmentation mask M_S) as follows:

$$\begin{aligned} O &= SegSwapper_{Seg(M_S)}(S, R') \\ &= S \cdot Seg(M_S) + R' \cdot (1 - Seg(M_S)). \end{aligned} \tag{4}$$

Tunable anonymity: To provide tunable anonymity and explore the best design tradeoffs, we consider both the impact of using different mask areas, each capturing different sets of identity-related features, and how the faces that are swapped are generated.

5.1 Example results

We next demonstrate the tunability of the approach. First, Tab. 1 shows the identity distances when swapping different example areas. Here, we show both the mean and standard deviation over 30,000 example images in CelebAMask-HQ dataset [19]. Second, Fig. 9 shows visual example results when masking on different areas of the face (top row). Overall, we have found that the approach provides excellent control of the level of anonymization to be achieved and that the results nicely address the shortcomings of only doing swapping in latent space, making it more attractive for videos and other contexts that place stricter requirements on preserving specific facial features and/or the background.

Table 1: Mean and standard deviation of identity distance when swapping a specific area in the face.

	Base	Eyes	Eyes + Nose	Eyes + Nose + Mouth
ID distance	1.25 ±0.10	0.28 ± 0.08	0.72 ± 0.14	0.79 ± 0.14

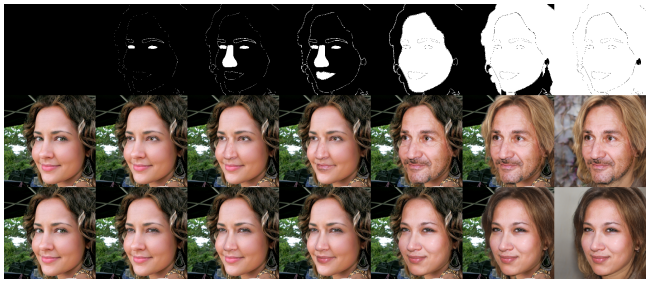


Figure 9: Example results when masking on different areas of the face (top row). Masked area from left to right: (1) none, (2) eyes only, (3) eyes+nose, (4) eyes+nose+mouth, (5) facial area, (6) facial area + hair, and (7) full masking.

The main drawback of the approach is that it can result in a mismatch of lighting between the randomly generated face and the original face. While we have found that most (but not all) such cases can be nicely corrected by a match color process, such a process requires additional care. We have also found a few cases when the segmentation mask was not well parsed, highlighting that the quality of the preservation depends on the accuracy of current state-of-the-art face parsers. In the next section we show how these results can be used to automate the face generation process.

6 LATENT SWAPPER

In previous sections, we proposed and evaluated two basic and easy-to-use methods that help anonymize the identity in latent space and pixel space. Both methods avoid the need to train big GAN models. This property is attractive since training large GANs are known to be complex and expensive (limiting who can train them). The models also provide desirable results for complementing problems. However, both approaches have some shortcomings that we address next. In particular, we present a method to automate the process of anonymizing faces in latent space, while obtaining similar results to the segmentation-based results.

6.1 High-level approach

To achieve our aim, we build a model that anonymizes a face in latent space using a ground truth built upon the results from our segmentation mask approach. The input of the model is latent codes of a real face, and the model is trained to find an α-mask that pushes the identity sufficiently far away from source identity that it is not recognized using the identification threshold of the FRS.

As illustrated in Fig. 10, we divide the model into three submodules: (1) a coarse attributes swapper, (2) an identity swapper,

and (3) a fine attributes swapper. Since the middle layers (see Sec. 3) hold most of the identity-related information, the loss function used during training gives most weight to the second module.

6.2 The loss function

We use the results of the segmentation mask swapper from Sec. 5 as the expected ground truth for the latent swapper model. This provides an anonymized face that shares the exact background and non-identity information but has a different identity. During the training process, we use two inputs to the model: the source latent code $L_S \in W$ and the latent code L_R of a random face $R = \mathcal{P}(L_R)$. Finally, we minimize the following loss so that it produces an identity mask α in latent space:

$$\mathcal{L} = \text{argmin}^{m\times n}_{\alpha\in(0,1)} \lambda_{L_2} |L_{\hat{S}} - t_{truth}| - \lambda_{ID} \mathcal{L}_{ID}(G(L_{\hat{S}}), S), \quad (5)$$

where $L_{\hat{S}} = \alpha \cdot L_S + (1-\alpha) \cdot L_R$; $m \times n$ is the size of latent codes in W; λ_{L2} and λ_{ID} are the lambda factor of the $L2$ latent loss and the cosine similarity between the identity of $G(L_{\hat{S}})$ and S; t_{truth} is the ground truth for the output of the model which is latent code of a target face T and $\mathcal{L}_{ID}(.,.)$ is the identity loss.

The second part of the objective function (eqn. (5)) ensures that the anonymized face $G(L_{\hat{S}})$ and S are further away in terms of identity distance. This is in fact the objective of the whole anonymization process. We see that the identity distance between S and output $G(L_{\hat{S}})$ gradually decrease due to there being pairs of S and R in the training dataset that are too close in term of identity distance. Furthermore, $Dist_{ID}(S,R) < Dist_{ID}(S,T)$, since during the process generating T, we use some components from S both in latent space and pixel space. Finally, we calculate the identity loss as:

$$\mathcal{L}_{ID}(S,\hat{S}) = C(F(G(L_{\hat{S}})), F(S)), \quad (6)$$

where $C(.,.)$ is the cosine similarity function and $F(.)$ is a facial recognition model that produces identity embedding vectors.

6.3 Training and example results

We used Arcface [4] as the facial recognition model during the training. Our model is trained on CelebAMask-HQ dataset [19] with 30,000 facial images in total, divided into training and testing sets by factor 90:10. Before the training process, we prepare the ground truth using the annotated segmentation mask provided in CelebAMask-HQ. Random faces are generated using 30,000 random vectors in latent space $\mathcal{Z}$ projected to latent space W, and then passed through the generator to get the random faces. The process of swapping identity in the random face set and blending them into the source face is described in Sec. 5. The swapper network has a similar structure to the mapper in StyleGAN but has only 4 fully connected layers. The input size is also double the size of latent code in $\mathcal{Z}$ so that it can take both some source latent codes L_S and latent codes L_R of random faces as input. Another important difference is that the swapper will pass (or put low weights) to layers that have low identity disentanglement, including layers [0,4] and [12,17]. Passing through these layers significantly boosts the network's convergence time. Finally, we train our model with hyperparameters $\lambda_{L2} = 1$, $\lambda_{ID} = 0.1$, and learning rate 0.1.

Fig. 11 shows example results. The results highlight that the mapper produces results (bottom row) where the identity has been pushed towards the semi-randomly generated face (second row) while preserving many of the features of the source face (top row). We next evaluate the performance of our system (Sec. 7) and compare the results with those produced by related works (Sec. 8).

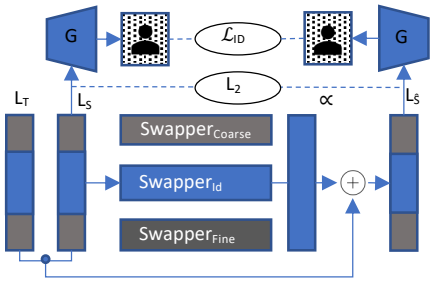


Figure 10: Network architecture of the latent swapper

Figure 11: Example result using the latent swapper. Source face (top), random face with same face mask and desirable distance (middle), and output face (bottom). Attributes to preserve: age, gender, face expressions, glasses.

7 EVALUATION

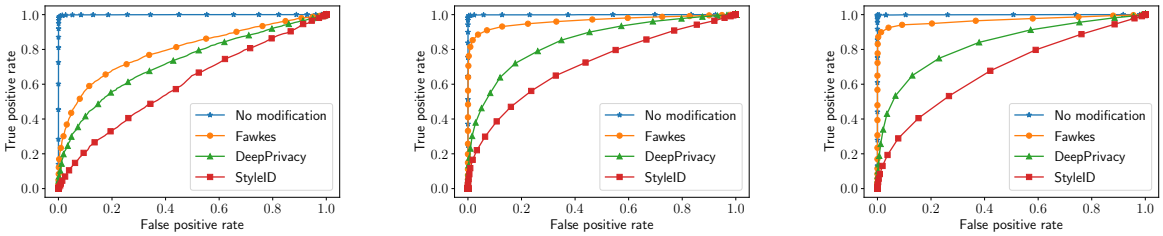
7.1 Privacy vs. facial recognition system (FRS)

We first evaluate StyleID's capability to protect the anonymized face against facial recognition and compare the results we achieve (here represented using the latent swapper) with the performance of other anonymization and de-identification methods. To prevent the "data leakage" problem, we use another dataset for evaluation. For privacy evaluation against FRS we use the LFW dataset [13]. The LFW is a standard dataset commonly used for evaluating the Facial Recognition models. The dataset includes 12,433 images of 5,749 identities and comes with a protocol to evaluate the facial recognition accuracy. We report results for three metrics.

1) Distance-based comparison: The first metric is the Euclidean distance between facial embeddings of the original source faces and the corresponding anonymized faces. To calculate the embeddings we use three state-of-the-art facial recognition models: Facenet [34], Arcface [4], and CurricularFace [14]. Tab. 2 presents results using this metric. Here, we show the average identity distance before and after applying different anonymization/de-identification techniques (plus/minus the standard deviation). As a reference, we also include a baseline column that contains the mean and standard

Table 2: Comparison of identity distance before and after applying different anonymization/de-identification methods.

FR model	Baseline	StyleID	DeepPrivacy [15]	k-same [28]	AnonFACES [18]	Fawkes [35]
FaceNet	1.03 ± 0.08	1.18 ± 0.08	1.2 ± 0.32	0.89 ± 0.12	0.98 ± 0.11	0.65 ± 0.08
ArcFace	1.19 ± 0.070	1.35 ± 0.11	1.21 ± 0.40	0.82 ± 0.13	0.97 ± 0.09	0.62 ± 0.06
CurricularFace	1.22 ± 0.13	1.29 ± 0.08	1.29 ± 0.42	0.98 ± 0.20	0.10 ± 0.11	0.72 ± 0.13



(a) FaceNet. **(b) ArcFace.** **(c) CurricularFace.**

Figure 12: ROC curves of different face embedding models on LFW benchmark.

Table 3: Area under the ROC curve (AUC) after applying different anonymization/de-identification methods.

FR model	Baseline	StyleID	DeepPrivacy [15]	Fawkes [35]
FaceNet	0.9994	0.6011	0.7291	0.7983
ArcFace	0.9994	0.7127	0.8465	0.9636
CurricularFace	0.9994	0.6805	0.8310	0.9684

Table 4: Accuracy on LFW benchmark when applying different anonymization/de-identification methods.

FR model	Baseline	StyleID	DeepPrivacy [15]	Fawkes [35]
FaceNet	0.9935	0.5755	0.6775	0.7347
ArcFace	0.9955	0.6599	0.7718	0.9238
CurricularFace	0.9981	0.6325	0.7538	0.9411

deviation of the pairwise Euclidean distance between random pairs of faces (belonging to different identities) in the evaluation dataset.

To avoid the risk of re-identification, it is desirable that the distance is equal or larger than the baseline. Both our framework and DeepPrivacy [15] achieve distances well above this baseline for all three face recognition tools. The very good performance using our framework when evaluated against ArcFace can be explained by our SwapperNet being optimized using the ArcFace identity embedding during the training process. Another noticeable difference is that DeepPrivacy has much higher standard deviation than us. One reason for this difference is that their in-painting method has some levels of entanglement with the surrounding context, while there is no mechanism to ensure the newly generated face is random. We observed this behavior while inputting two relatively similar faces and the output results are roughly the same.

The average distance of the two face averaging methods (k-same [28] and AnonFACES [18]) are both well below the baseline. This is because these methods average faces in a cluster into a single face. Finally, Fawkes [35] has the lowest score using this metric, suggesting that its privacy filter sees the biggest chance of re-identification. Part of the reason may be that Fawkes has not been optimized against state-of-the-art facial recognition models trained with triplet loss such as those that we are using.

2) Receiver operating characteristic (ROC) comparison: A popular approach to evaluate how effective anonymization methods are against an FRS is to use ROC curves. We use LFW's benchmark protocol [13], which requires multiple face images per identity, and exclude AnonFACE [22] and k-same [28] (assumed one image per identity) from this evaluation. Figs. 12a, 12b and 12c show head-to-head comparisons of our method when using FaceNet, ArcFace, and CurricularFace. In all cases, we compare against both Fawkes [35] and DeepPrivacy [15]. As a baseline we also include the use of the original dataset ("no modification" shown in blue). StyleID (red) significantly outperforms Fawkes (orange) and DeepPrivacy (green), as demonstrated by its ROC curves consistently being closer to the diagonal. While Fawkes (orange) provides some protection against FaceNet, it provides very limited protection against the two newer face embedding models. DeepPrivacy (green) consistently outperforms Fawkes but has less attractive tradeoff curves than StyleID. Supporting these ROC results, we provide additional quantitative metrics: Area Under the Curve (AUC) shown in Tab. 3 and the Accuracy shown in Tab. 4. Here, lower values are better.

3) Rank-based comparison: Third, we used the average of the true rank in the gallery dataset. Here, the ranks are calculated by passing the anonymized faces through an FRS built on Microsoft face API [27]. Specifically, the evaluation was carried out as follow: (1) LFW is split into gallery and probe sets. (2) The images in the gallery, with their identify labels, are used for building the facial recognition. (3) For each probe image passing through the facial recognition model, the rank of the true identity is recorded. Here, we say that a probe image p_i achieves rank k when there are $k-1$ other identities that have lower identity distance than the identity in the probe image p_i. (4) The same procedure as in step 3 is applied on anonymized/de-identified images in the probe set. Tab. 5 reports the average and standard deviation over the whole probe set.

Similar to the first evaluation, we also include a baseline where images from the same person are passed through the FRS. The result shows that the Microsoft face's API is highly accurate in recognizing faces in the LFW dataset. We observe a similar trend to the identity distance metric. StyleID achieves the best results, slightly outperforming DeepPrivacy [15]. Both frameworks achieve an average rank of around 1,000 using a dataset with around 5,000 identities in the gallery. This is approximately 6-10 times better than k-Same and AnonFACES, and 20 times better than Fawkes.

While the rank-based results may suggest that our performance is only slightly better than DeepPrivacy [15], we note that our framework typically is much better at preserving facial features, and provides more appealing and natural looking results (cf. Fig. 16).

Table 5: Comparison of identification rank before and after applying different anonymization/de-identification methods

	Baseline	StyleID	DeepPrivacy [15]	k-same [28]	AnonFACES [18]	Fawkes [35]
Identification rank	1.02 ± 0.08	1,027.21 ± 25.68	925.43 ± 107.54	107.32 ± 57.47	153 ± 32.51	5.29 ± 2.68

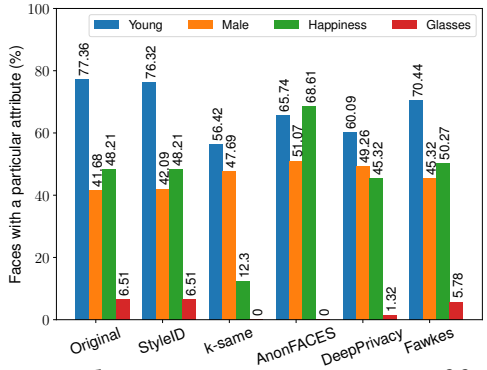


Figure 13: Attribute preservation. Percentage of faces with selected example attributes before (Original) and after applying different methods (closer to "Original" is better).

Figure 14: Controlling gender attribute in latent space. Top row: original faces, second row: random target faces, third rows: face with corrected gender attribute, bottom row: results by swapping layers (5,6,7)

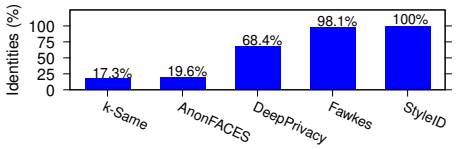


Figure 15: Relative number of identities in the face image dataset after the anonymization/de-identification process.

7.2 Attributes preservation

For high utility, it is important to preserve both the attributes of individual faces and the distribution of attributes in the dataset.

Dataset properties: We first consider how well the methods preserve the distribution of example properties in the dataset. This experiment is conducted on the CelebA dataset (we use the labelled attributes in the dataset as the attributes before anonymization) and uses Microsoft's Face API as attribute extractor after anonymization. For this evaluation, we selected four example attributes of interest: "Young" (defined as younger than 30 here), "Male", "Happiness", and "Glasses". The first three attributes are popular attributes in the dataset and the last is a minority attribute added to demonstrate that the anonymizer works well with special attributes too.

Fig. 13 shows the distributions of the four example attributes after applying the different methods on the original image dataset. We see that our framework provides a good match with the original (bottom set). Of the others, only Fawkes [35] is able to preserve the minority attribute "Glasses". This is perhaps not surprising since Fawkes applies a very light privacy filter. The other techniques perform poorly, often averaging out many attributes in individual faces, resulting in minority (attribute such as glasses often disappearing).

Controlling attributes of individual faces: While our default solutions are good at preserving many facial attributes, further control can easily be added. To show this, note that there exist many tools for controlling non-identity related attributes, including StyleSpace [43], GANSpace [9], InterfaceGAN [36], StyleCLIP [31]. In Fig. 14, we demonstrate a case where some randomly generated target faces (second row) have different gender compared to the original faces (first row). The gender is corrected in the third row by using global direction of GANSpace (only for faces with incorrect gender). In this case, our goal is to control the gender attribute of the target. Referring back Sec. 3.8, the analysis suggested that swapping layers (5,6,7) will result in more attributes captured from the target. We therefore swap these layers of the latent codes in the first row, to the ones in the third row. The swapped results are shown in the fourth row in which the gender attribute is preserved.

The properties outlined here are important for anonymization since they allow us to add further prevention of re-identification attacks in which a set of meta-information about the face may be used to re-identify a person. For example, the capability of the anonymization method to manipulate a given attribute is valuable for concealing minority identities. While we have demonstrated that we can successfully preserve these properties, it is trivial to change some subset of minority properties if this would be desired (e.g., due to an attacker having knowledge about all individuals' attribute values) by manipulation in latent space. This allows us to easily further enhance the level of anonymization (e.g., by applying l-diversity or t-closeness on attributes), and by giving a constraint on the attributes before the anonymization process.

7.3 Identity diversity

To ensure high utility, it is also important to preserve the number of identities in the dataset. For this evaluation, we extract a sample image for each of the 10,777 identities in CebebA [23]. After the anonymization/de-identification process of each method, we extract face embeddings using the CurricularFace model, cluster the embedding using k-means [24], and use the number of clusters as a proxy for the number of identities in the final dataset. Fig. 15 shows the percentage of identities compared to in the original dataset.

As desired, StyleID preserves the number of identities. In contrast, the number of identities decreases to below 20% when using k-same and AnonFACES with $k = 5$ (as suggested in those papers). This clearly is a weakness of k-anonymity-based methods. Another

interesting observation is that the generated identities by DeepPrivacy have low diversity, in the sense that different original faces can share visually similar anonymized faces. As a result, it sees a 32% drop in the number of unique identities. Of the other techniques only Fawkes does a good job preserving the number of identities.

8 COMPARISONS WITH RELATED WORKS

8.1 Related work

Early anonymization works were primarily based on the idea of k-Anonymity [38]. These works often use different embedding spaces to find k similar faces [5, 26, 37], and replace them by the average face of the cluster. This helps reduce the chance to re-link an original face to at most $1/k$. For example, k-same-pixel [28] and k-same-M are based on active appearance model [8], while AnonFACES [18] recently showed how faces can be averaged in the latent space of a GAN. The downsides of this approach are that it reduces the number of identities by a factor of k and it does not consider attributes. We instead focus on privacy in the FRS context and derive privacy/utility metrics from the machine learning field.

Recent works have instead focused on identity de-identification with the sole purpose of concealing identity in facial images/videos. For example, [15, 21, 25, 30, 42] all propose custom GAN models for image de-identification. However, they also all share the same weakness in that they lack naturalness. This is not a surprise since the facial domain is one of the most challenging domains for GAN models and the training process is complex and time consuming.

Gafni et al. [7] apply similar models to faceswap [6] (originally known as deepfakes) to de-identify video. This is one of a few solutions for videos. Unfortunately, there is no source code available and we could not re-produce the claimed results (but we can compare our visual outputs with theirs; see next section). Fawkes [35] also aims to conceal the identity in facial images but takes a different approach (adds a privacy filter to the facial images) and the objective is also slightly different (i.e., adding invisible noises to hide identity in feature space of convolutional neural network). This way, they try to fool some FRS:s without significant changes in the pixel space. The author provided the software and source codes so that anyone can test the results. However, on testing with real world image samples we found that this method is not effective against state-of-the art facial recognition models trained with triplet loss such as FaceNet [34], ArcFace [4], CurricularFace [14].

8.2 Visual comparison to the related work

Fig. 16 compares StyleID with the main related works that we could reproduce. Here, the first row shows example of random source images. For Fawkes (second row) we use a configuring of "–mode mid", meaning that the protection is at middle level of three options ("low", "mid", and "high"). With this method, a form of random noise is added to the source images. However, in contrast to the authors' claim of "imperceptible" noise, the noise appears visible in our test results. With DeepPrivacy [15] (third row) we typically observed that the facial area had low resolution and looked unnatural. Furthermore, we noted in Sec. 7.2, attributes such as facial expression, glasses, and eyes direction were often not preserved. AnonFACES [18] (fourth row) achieved similar image quality as StyleID (it is also based on the StyleGAN generator) but did not have control of the facial attributes, often resulting in uncontrolled changes in attributes such as pose, expression, gender, and age. The best performance was obtained with StyleID (fifth row), which achieved both high naturalness and nicely preserved the most important attributes (e.g., facial expressions, age, gender, glasses, etc.).

In addition, we selected to compare with CIAGAN [25] (good baseline among works sharing their code) and the work by Gafni et al. [7] (discussed above, provides among the most promising results we had seen, but do not share their code). Figs. 17 and 18 present comparisons with example results from these works. In all cases our approach better preserve the facial features and naturalness simultaneously as we provide as good or better protection against FRS:s (e.g., see evaluation results in Sec. 7 for comparison with [15]). Similar to our original ideas, Gafni et al. [7] use an identity loss to push the identity further away from original images. What we have found (and which is visible in their result) is that this can converge identities as "furthest away" may push results toward the same point in latent space. This is the reason almost all noses become similar (see Fig. 18). To solve this problem (and ensure high visual diversity) we integrated the generation of random faces with a similar segmentation mask and then built that into the ground truth of the training of our latent swapper. This allows our results to have a distinct look of a different identity. Furthermore, the amount of change can easily be adjusted in our case, as we do not require a lengthy re-training process (which [7] would require).

9 WIDER CONTEXT

9.1 More advanced attacker model

In our evaluation in Sec. 7, we assumed an adversary that uses one of three different state-of-the-art FRS:s to try to re-identify the original face. Here, StyleID was shown to significantly outperform prior works. A more advanced attacker may also have access to our anonymizer and may try to perform a type of parrot attack. Here, we assume a threat model where the attacker does not have access to the gallery set of the FRS but can create an arbitrary probe set, can make a large number of queries via the FRS's API, and can generate anonymized faces using the anonymizer (that can be feed to the FRS). For this scenario, our use of random vectors to generate random synthetic faces ensures that the attacker cannot easily reverse engineer the original faces through guessing. This would be trivial if the model was deterministic. Using a probabilistic function, the chance of the attacker guessing the correct anonymized face is as good as predicting the next random $z \in \mathcal{Z}$. While the subset of $z \in \mathcal{Z}$ that corresponds to real faces is substantially smaller than the size of $\mathcal{Z}$, the probability of guessing the next face being generated is still small, making this a non-trivial task. In this discussion we have ignored that the attacker may use the background to identify an image from the probe set. To protect also against such attacker modifications also to the background would be needed. This is outside the scope of this paper. In fact, to provide good utility we prefer to keep the background and hair in most cases.

9.2 Generated faces match real people

Using StyleGAN's face generator, we acknowledge that there is a risk that a generated face matches the face of a real person. This is an issue discussed in the GAN research community that follow-up

Figure 16: Comparison of our results to Fawkes [35], DeepPrivacy [15], AnonFACES [18]

Figure 17: Comparison CIAGAN [25].

Figure 18: Comparison Gafni et al. [7].

works should be aware of. However, we note that the possibility of the generated face matching a real person is similar to finding two people with the same facial identity. In practice, with 8 billion people alive today, having a matching identity with another person is today not seen as privacy concern.

9.3 Ethical statement

Synthetic data generated by GAN models and DeepFakes can be misused for harmful purposes such as impersonation and misinformation. Our work intends to protect facial identities and to benefit the machine learning community by enabling privacy-preserving collection and sharing of training data. We publish our source codes for transparency purposes and encourage the open-source community to build upon our work for the greater good. However, we also acknowledge that the work, as well as the related works in the field of synthetic data generation, could be misused. We therefore raise a warning and highly recommend having the consent of the people in the collected images/video and carefully consider the legal aspect before using the proposed techniques.

9.4 Practical applications

StyleID provides an effective way to anomymize/ de-identify image datasets. This helps protect individual's identity. This has several applications, including to anonymize faces before uploading images to social media platforms or to create anonymized facial image data that can be used for research and/or to improve the accuracy and robustness of machine learning models (e.g., face analysis/recognition models, facial image synthetic models, 3D facial reconstruction models) applicable for a broad range of topics. StyleID may be particularly valuable when wanting to simultaneously preserve contextual information and respect the privacy of bystanders [10] incidentally captured in photos. The insight from our work on identity disentanglement can also help develop DeepFake detection algorithms (e.g., leveraging our approach to swap the identity of a face to many random identities can add fidelity to their training datasets). Finally, StyleID can easily be combined with other frameworks on semantic image editing in ways that could benefit fields such as photography, cinematic, and gaming.

10 CONCLUSION

We have presented the design and evaluation of StyleID, a feature-preserving anonymization framework that protects the individual's identity, while preserving as many characteristics of the original faces in the image dataset as possible. StyleID is based on the insights derived through the definition of a new identity disentanglement metric and the incremental development (and evaluation) of three complementing disentanglement methods that each build upon the insights and results of the prior methods. StyleID provides tunable privacy, efficacy, and is shown to outperform current state-of-the-art solutions. The high utility of datasets that can be generated using StyleID is believed to be an enabler for more and better training data for machine learning models that need to operate in contexts where there are people. Our code can be found here: https://github.com/minha12/StyleID.

ACKNOWLEDGMENTS

This work was supported by the Wallenberg AI, Autonomous Systems and Software Program (WASP) funded by the Knut and Alice Wallenberg Foundation.

REFERENCES

[1] Rameen Abdal, Yipeng Qin, and Peter Wonka. 2020. Image2stylegan++: How to edit the embedded images?. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 8296–8305.
[2] deepfakes. 2019. faceswap. https://github.com/deepfakes/faceswap. Accessed: April 2020.
[3] Jiankang Deng, Jia Guo, Xue Niannan, and Stefanos Zafeiriou. 2019. ArcFace: Additive Angular Margin Loss for Deep Face Recognition. In *CVPR*.
[4] Jiankang Deng, Jia Guo, Niannan Xue, and Stefanos Zafeiriou. 2019. Arcface: Additive angular margin loss for deep face recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 4690–4699.
[5] Liang Du, Meng Yi, Erik Blasch, and Haibin Ling. 2014. GARP-face: Balancing privacy protection and utility preservation in face de-identification. In *IEEE International Joint Conference on Biometrics*. IEEE, 1–8.
[6] Faceswap. 2017. Github project, https://github.com/deepfakes/faceswap. https://github.com/deepfakes. (2017).
[7] Oran Gafni, Lior Wolf, and Yaniv Taigman. 2019. Live face de-identification in video. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 9378–9387.
[8] Ralph Gross, Latanya Sweeney, Fernando De la Torre, and Simon Baker. 2006. Model-based face de-identification. In *2006 Conference on computer vision and pattern recognition workshop (CVPRW'06)*. IEEE, 161–161.
[9] Erik Härkönen, Aaron Hertzmann, Jaakko Lehtinen, and Sylvain Paris. 2020. Ganspace: Discovering interpretable gan controls. *arXiv preprint arXiv:2004.02546* (2020).
[10] Rakibul Hasan, David Crandall, Mario Fritz, and Apu Kapadia. 2020. Automatically detecting bystanders in photos to reduce privacy risks. In *IEEE Symposium on Security and Privacy (S&P)*. 318–335.
[11] Kashmir Hill. 2020. The secretive company that might end privacy as we know it. *The New York Times* 18 (2020), 2020.
[12] Andrew G Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam. 2017. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861* (2017).
[13] Gary B Huang, Marwan Mattar, Tamara Berg, and Eric Learned-Miller. 2008. Labeled faces in the wild: A database forstudying face recognition in unconstrained environments. In *Workshop on faces in'Real-Life'Images: detection, alignment, and recognition*.
[14] Yuge Huang, Yuhan Wang, Ying Tai, Xiaoming Liu, Pengcheng Shen, Shaoxin Li, Jilin Li, and Feiyue Huang. 2020. Curricularface: adaptive curriculum learning loss for deep face recognition. In *proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 5901–5910.
[15] Håkon Hukkelås, Rudolf Mester, and Frank Lindseth. 2019. Deepprivacy: A generative adversarial network for face anonymization. In *International Symposium on Visual Computing*. Springer, 565–578.
[16] Tero Karras, Samuli Laine, and Timo Aila. 2019. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 4401–4410.
[17] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. 2019. Analyzing and improving the image quality of stylegan. *arXiv preprint arXiv:1912.04958* (2019).
[18] Minh-Ha Le, Md Sakib Nizam Khan, Georgia Tsaloli, Niklas Carlsson, and Sonja Buchegger. 2020. AnonFACES: Anonymizing Faces Adjusted to Constraints on Efficacy and Security. In *Proceedings of the 19th Workshop on Privacy in the Electronic Society*. 87–100.
[19] Cheng-Han Lee, Ziwei Liu, Lingyun Wu, and Ping Luo. 2020. Maskgan: Towards diverse and interactive facial image manipulation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 5549–5558.
[20] Ninghui Li, Tiancheng Li, and Suresh Venkatasubramanian. 2007. t-closeness: Privacy beyond k-anonymity and l-diversity. In *IEEE International Conference on Data Engineering*. 106–115.
[21] Tao Li and Lei Lin. 2019. Anonymousnet: Natural face de-identification with measurable privacy. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*. 0–0.
[22] Tao Li and Lei Lin. 2019. Anonymousnet: Natural face de-identification with measurable privacy. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*. 0–0.
[23] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. 2015. Deep Learning Face Attributes in the Wild. In *Proceedings of International Conference on Computer Vision (ICCV)*.
[24] Stuart Lloyd. 1982. Least squares quantization in PCM. *IEEE transactions on information theory* 28, 2 (1982), 129–137.
[25] Maxim Maximov, Ismail Elezi, and Laura Leal-Taixé. 2020. Ciagan: Conditional identity anonymization generative adversarial networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 5447–5456.
[26] Lily Meng, Zongji Sun, Aladdin Ariyaeeinia, and Ken L Bennett. 2014. Retaining expressions on de-identified faces. In *2014 37th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*. IEEE, 1252–1257.
[27] Microsoft. 2021. Microsoft's Face API. https://azure.microsoft.com/en-us/services/cognitive-services/face/. Accessed: Nov 2021.
[28] Elaine M Newton, Latanya Sweeney, and Bradley Malin. 2005. Preserving privacy by de-identifying face images. *IEEE transactions on Knowledge and Data Engineering* 17, 2 (2005), 232–243.
[29] Yuval Nirkin, Yosi Keller, and Tal Hassner. 2019. Fsgan: Subject agnostic face swapping and reenactment. In *Proceedings of the IEEE/CVF international conference on computer vision*. 7184–7193.
[30] Yi-Lun Pan, Min-Jhih Haung, Kuo-Teng Ding, Ja-Ling Wu, and Jyh-Shing Jang. 2019. K-same-siamese-gan: K-same algorithm with generative adversarial network for facial image de-identification with hyperparameter tuning and mixed precision training. In *2019 16th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*. IEEE, 1–8.
[31] Or Patashnik, Zongze Wu, Eli Shechtman, Daniel Cohen-Or, and Dani Lischinski. 2021. Styleclip: Text-driven manipulation of stylegan imagery. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2085–2094.
[32] Ivan Perov, Daiheng Gao, Nikolay Chervoniy, Kunlin Liu, Sugasa Marangonda, Chris Umé, Mr Dpfks, Carl Shift Facenheim, Luis RP, Jian Jiang, et al. 2020. Deepfacelab: A simple, flexible and extensible face swapping framework. *arXiv preprint arXiv:2005.05535* (2020).
[33] Elad Richardson, Yuval Alaluf, Or Patashnik, Yotam Nitzan, Yaniv Azar, Stav Shapiro, and Daniel Cohen-Or. 2021. Encoding in style: a stylegan encoder for image-to-image translation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2287–2296.
[34] Florian Schroff, Dmitry Kalenichenko, and James Philbin. 2015. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 815–823.
[35] Shawn Shan, Emily Wenger, Jiayun Zhang, Huiying Li, Haitao Zheng, and Ben Y Zhao. 2020. Fawkes: Protecting privacy against unauthorized deep learning models. In *29th {USENIX} Security Symposium ({USENIX} Security 20)*. 1589–1604.
[36] Yujun Shen, Ceyuan Yang, Xiaoou Tang, and Bolei Zhou. 2020. Interfacegan: Interpreting the disentangled face representation learned by gans. *IEEE transactions on pattern analysis and machine intelligence* (2020).
[37] Zongji Sun, Li Meng, and Aladdin Ariyaeeinia. 2015. Distinguishable de-identified faces. In *2015 11th IEEE International Conference and Workshops on Automatic Face and Gesture Recognition (FG)*, Vol. 4. IEEE, 1–6.
[38] Latanya Sweeney. 2002. k-anonymity: A model for protecting privacy. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems* 10, 05 (2002), 557–570.
[39] Yuri Viazovetskyi, Vladimir Ivashkin, and Evgeny Kashin. 2020. Stylegan2 distillation for feed-forward image manipulation. In *European Conference on Computer Vision*. Springer, 170–186.
[40] Andrey Voynov and Artem Babenko. 2020. Unsupervised discovery of interpretable directions in the gan latent space. In *International Conference on Machine Learning*. PMLR, 9786–9796.
[41] Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. 2018. High-resolution image synthesis and semantic manipulation with conditional gans. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 8798–8807.
[42] Yifan Wu, Fan Yang, Yong Xu, and Haibin Ling. 2019. Privacy-protective-GAN for privacy preserving face de-identification. *Journal of Computer Science and Technology* 34, 1 (2019), 47–60.
[43] Zongze Wu, Dani Lischinski, and Eli Shechtman. 2021. Stylespace analysis: Disentangled controls for stylegan image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 12863–12872.
[44] Sharon Zhou, Mitchell Gordon, Ranjay Krishna, Austin Narcomey, Li F Fei-Fei, and Michael Bernstein. 2019. Hype: A benchmark for human eye perceptual evaluation of generative models. *Advances in neural information processing systems* 32 (2019).

A CORRELATION BETWEEN IDENTITY AND INDIVIDUAL ATTRIBUTES

The latent space of StyleGAN is among one of the most disentangled latent spaces in the sense that it allows manipulation of a particular attribute or a set of attributes while keeping the remaining

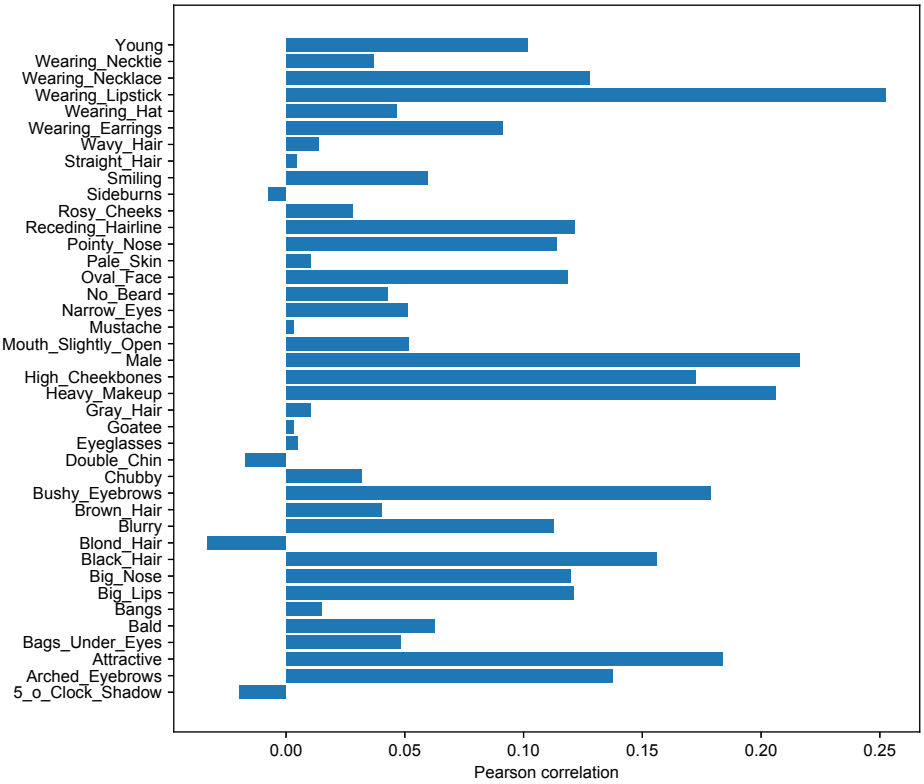


Figure 19: Correlation of change in identity distance and L1 distance of attributes when swapping layer [5,6,7]

attributes un-changed. As shown in the related works [9, 31, 36, 43] this allows meaningful manipulation of the attributes. However, when it comes to identity, we observed that there is a degree of correlation between identity and attributes. Based on our analysis of swapping layers and channels, we have found that there are different combinations of layers or channels that can result in the best identity disentanglement score. This is perhaps not surprising since different people have different characterizing features and the identity disentanglement score combines the score over all attributes.

In this appendix, we look closer at the correlation between the change in identity distance and the changes in individual attributes for a number of example transformations. Here, we chose to swap layers (5,6,7); i.e., the consecutive layers that were found to most effectively shift the identity to the target while maintaining a high disentanglement score. Of particular interest is the correlation in the change in the identity distance (L2/Euclidean distance) and the changes in the attribute score (which is the confidence value of the attribute classifier $AttrNet$). Here, we measure the change of the individual attribute score using the L1 distance (as the single score of an attribute is a scalar value).

Fig. 19 shows the Pearson correlation of the change in identity with all 40 attributes of CelebA dataset as seen for this particular example experiment. The six attributes with the highest correlation to the identity were: “Wearing Lipstick", “Male", “Heavy Makeup", “Attractive", “Bushy Eyebrows", and “High Cheekbones". While some of these attributes (e.g., “Wearing Lipstick" highest correlated to identity here) may appear non-intuitive at first, most of these top-ranked attributes help distinguish the gender of a client (e.g., “Wearing Lipstick" and “Heavy Makeup" may traditionally be more likely associated with females, whereas “Male" and “Bushy Eyebrows" may be better at indicating that a person is more likely to be male). These clues are in line with our observation that gender is highly correlated with the identity and one of the attributes most entangled to the identity.

Some other attributes that also have some degree of correlation to the identity include “Young", “Wearing Necktie", “Black Hair", “Big Nose", etc. In contrast, attributes such as “Mustache", “Eyeglasses", “Blond Hair" have a low or inverse correlation to the identity.