der Bundeswehr
Universität München

Deep Learning-Based Approaches to Face De-Identification with Data Utility Preservation

Andreas Leibl
Neubiberg 2025

Vollständiger Abdruck der von der Fakultät für Informatik der Universität der Bundeswehr München zur Erlangung des akademischen Grades eines

Doktors der Naturwissenschaften (Dr. rer. nat.)

angenommenen Dissertation.

Gutachter/Gutachterin:
1. Univ.-Prof. Dr.-Ing. habil. Helmut Mayer
2. Univ.-Prof.'in Dr. Marta Gomez-Barrero

Die Dissertation wurde am 12.05.2025 bei der Universität der Bundeswehr München eingereicht und durch die Fakultät für Informatik am 26.06.2025 angenommen. Die mündliche Prüfung fand am 24.07.2025 statt.

Abstract

In this thesis, we present two novel approaches to de-identify visual data. They leverage Generative Adversarial Networks and Diffusion Models, two recently developed generative deep learning techniques, to replace real faces with synthetically generated surrogates. The advantage of this approach over traditional anonymization with pixelization or blurring is that it can retain data utility for downstream tasks that require processing the human face.
Our first approach, *DetailedPrivacy*, can preserve expression, pose and gaze of individual faces but can only be applied to images and videos containing relatively large faces with little occlusion.
Our second approach, on the other hand, *StablePrivacy*, can be applied to more complex scenes and alters faces more drastically. It achieves state-of-the-art protection against identification by deep learning-based face recognition methods. Moreover, it retains the utility necessary for training deep learning-based face detection models on anonymized data better than all other approaches we evaluated.

Zusammenfassung

Im Rahmen dieser Dissertation werden zwei neuartige Verfahren zur De-Identifizierung visueller Daten vorgestellt. Diese nutzen Generative Adversarial Networks und Diffusion Models, zwei kürzlich entwickelte generative Verfahren des Deep Learning, um reale Gesichter durch synthetisch generierte Stellvertreter zu ersetzen. Der Vorteil dieses Ansatzes gegenüber herkömmlicher Anonymisierung mittels Verpixelung oder Bildglättung besteht darin, dass die Daten ihre Eignung für spätere Anwendungen, die eine Verarbeitung des Gesichts erfordern, erhalten.
Der erste Ansatz *DetailedPrivacy* bewahrt den Gesichtsausdruck, die Kopfausrichtung und den Blick einzelner Gesichter. Allerdings ist er auf Bilder und Videos mit relativ großen, kaum verdeckten Gesichtern beschränkt.
Der zweite Ansatz *StablePrivacy* eignet sich hingegen auch für komplexere Bildszenen. Er verändert die Gesichter stärker und erreicht somit einen Schutz vor der Wiedererkennung durch Deep-Learning-basierte Gesichtserkennungsmodelle, der dem Stand der Technik entspricht. Darüber hinaus bewahrt StablePrivacy die Eignung der anonymisierten Bilder für das Training von Modellen für die Gesichtslokalisierung besser als alle anderen evaluierten Ansätze.

Acknowledgments

First and foremost, I want to thank my supervisor, Prof. Helmut Mayer, for his support and for giving me the opportunity to pursue my PhD. Your guidance helped me overcome the difficulties along my path and your feedback made this work what it is today.
Secondly, I want to thank my friends and colleagues for making work fun, insightful discussions and helping me out when deadlines came up.
Finally, and especially, I want to thank my parents and my partner, Antonia, for their unconditional and inexhaustible support.

Contents

Chapter 1

Introduction

The recent fast-paced progress in deep learning-based computer vision relies heavily on the availability of large-scale image datasets [Sun et al., 2017; Everingham et al., 2008; Deng et al., 2009; Lin et al., 2014; Kuznetsova et al., 2020; Schuhmann et al., 2022]. While the potential benefits of this technological progress to society are undeniable, the dependence on these datasets comes with risks to personal privacy [Paullada et al., 2020; Birhane and Prabhu, 2021], as they often contain a large percentage of images depicting people. Even ImageNet [Deng et al., 2009], a classification dataset that only contains three categories that directly concern humans (scuba diver, bridegroom, and baseball player), consists of at least 17 % images featuring people. These can often be easily linked to an individual's real identity through reverse image search, which raises serious concerns about misuse [Birhane and Prabhu, 2021].

To prevent the exploitation of such data around the world, privacy regulations, such as the General Data Protection Regulation (GDPR) in Europe [European Union, 2016], the California Consumer Privacy Act and California Privacy Rights Act [California Legislative Counsel, 2018] or the Australian Privacy Act [Australian Government] and the Australian Privacy Principles [Office of the Australian Information Commissioner], protect citizens. The usage of protected data, even when publicly available, can lead to large fines even for companies operating outside the jurisdiction of these countries. For example, in the Clearview AI case, a US company was fined for violating the GDPR because it used publicly available biometric data of EU citizens [Jung and Kwon, 2024]. However, stringent regulations concerning the processing,

storage, and sharing of privacy-sensitive visual data, e.g., faces, can be challenging for companies and researchers, hampering productivity. A seemingly simple solution is to anonymize images by obfuscating relevant areas. Thus, researchers started pixelizing or blurring faces or the entire body in many publicly available datasets [Uittenbogaard et al., 2019; Caesar et al., 2020; Piergiovanni and Ryoo, 2020; Yang et al., 2021].

1.1 Problem Statement

For certain tasks, pixelizing or blurring is a valid strategy, as the effect of anonymization is relatively minor. For instance, Yang et al. [2021] demonstrated that blurring faces in ImageNet only decreased the top-5 accuracy by 0.4 % on average for 15 different classification models. On the other hand, the authors also found that the effect is much more significant for objects that typically appear near faces, e.g., masks (8.71 %) or harmonicas (8.93 %). Moreover, there are many tasks that require directly analyzing images of humans, such as face segmentation, action recognition, face detection or face recognition. In these cases, simple anonymization of datasets can lead to a severe reduction in data utility, rendering them unsuitable for scientific research or commercial development. For example, a deep-learning model trained for face detection on heavily pixelated faces might be more tuned towards localizing pixel boxes than real faces.
Ideally, the chosen de-identification approach would remove all privacy-sensitive content from the original images while preserving all other meaningful information. In practice, these two objectives usually conflict. The more the image is being altered to eliminate identifying features, the less utility is retained. Thus, there is a fundamental *privacy – data utility trade-off*.
One alternative to obfuscating the face that promises a better trade-off is to replace it with a synthetically generated surrogate, leveraging generative deep-learning techniques (cf. Section 2.5). Related work, such as by Maximov et al. [2020] or Hukkelås et al. [2019], which we will further discuss in Section 3.2.3, demonstrates the potential of this method.
Still, due to the inherent conflict between privacy and utility, preserving unneeded details can unnecessarily compromise privacy without adding

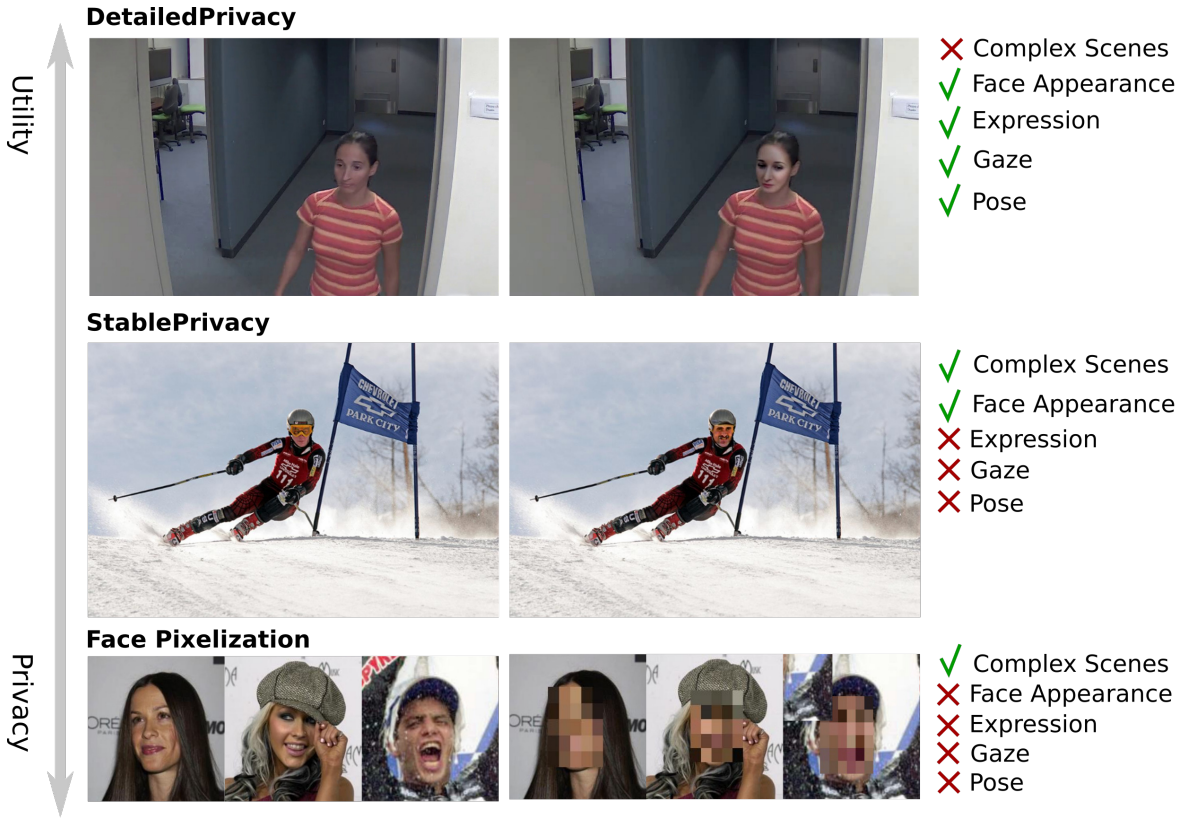


Figure 1.1: We have developed two novel approaches for face de-identification: *DetailedPrivacy* and *StablePrivacy*. Each has a different *privacy – data utility trade-off* influenced by task-specific requirements. Original images (left side) from Wong et al. [2011]; Yang et al. [2016]; Huang et al. [2008].

value. Consequently, as data utility is typically task-specific, it is important to tailor which information is maintained according to the requirements of the downstream use case (see Figure 1.1). To illustrate this point, let us reconsider training a face detection model on anonymized data. In this scenario, it is sufficient to retain generic human shape and texture. Beyond that, transferring the exact expression only risks leaking identifying features. Conversely, if the downstream task requires precise semantic understanding, it can be necessary to preserve such details, although it can negatively affect privacy.

Hence, in this thesis, we have developed two novel approaches to de-identification, optimized for different use cases. Our first approach, *DetailedPrivacy*, retains expression, pose and gaze of individual faces, but depends on detailed landmarks extracted from the original face, resulting in less effective privacy protection. On the other hand, our second approach, *StablePrivacy*,

does not depend on such landmarks and has, therefore, more freedom to completely alter the face, offering stronger privacy. Moreover, it can even be applied to complex scenes with small or heavily occluded faces, for which landmarks cannot be determined.

1.2 Thesis Outline

This thesis consists of the following chapters:

Chapter 2: Theoretical Background. Here, we describe the theoretical prerequisites for this thesis, beginning with the fundamentals of Artificial Neural Networks (ANNs). Next, we discuss Convolutional Neural Networks (CNNs), a special type of ANNs that is adapted to the image domain and go into detail about typical architectures. These networks constitute the backbones of the models we employ for face detection and face recognition, which we describe in the following sections. They are used to quantify data utility retention and privacy protection. The last section of this chapter covers deep learning-based image generation methods, which are essential for creating surrogate faces.

Chapter 3: Related Work: Face De-Identification with Utility Retention. In the related work chapter, we first describe the existing landscape of biometric privacy-enhancing techniques, focusing on synthesis-based image anonymization, the category to which our approaches belong. Additionally, we specify the datasets and evaluation strategies used for performance evaluation and introduce the state of the art, to which we will compare our approaches.

Chapter 4: Two Novel Approaches for Synthesis-Based Privacy Enhancement. This is the core chapter of this thesis and it contains the novel contributions. It describes in detail our approaches to synthesis-based face de-identification and provides an in-depth evaluation of privacy protection and utility retention.

Chapter 5: Summary and Outlook. In the conclusion, we summarize our contributions and give an outlook to possible future research.

Finally, the appendix contains a table of the mathematical notation and abbreviations used in this thesis.

Chapter 2

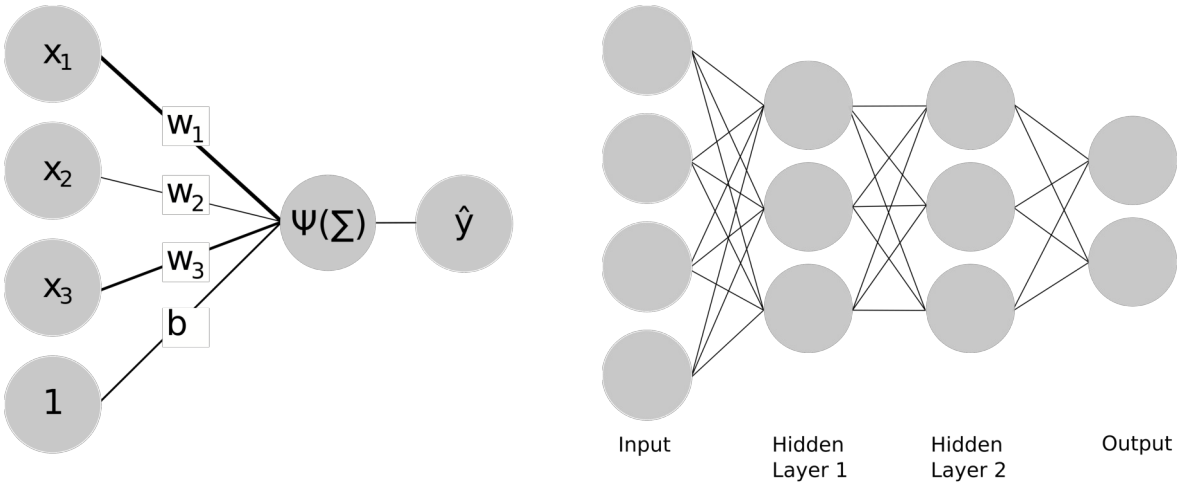
Theoretical Background

This chapter lays out the theoretical background for this thesis. As our research focuses on image de-identification strategies that leverage Artificial Neural Networks (ANNs), we begin by outlining their fundamentals in Section 2.1. This is followed by an in-depth discussion of Convolutional Neural Networks (CNNs), a variant of ANNs especially suitable for image data, which are the basis for most of the machine-learning methods used in this thesis (cf. Section 2.2). Next, we provide an overview of face detection in Section 2.3 and face recognition in Section 2.4, which serve as the primary methods for empirically evaluating data utility and privacy protection in de-identified datasets. Finally, we discuss deep learning-based image generation techniques, focusing on Generative Adversarial Networks (GANs) and diffusion models in Section 2.5, as they are essential components of the face de-identification approaches explored in this work.

2.1 Artificial Neural Networks

The fundamental objective of ANNs is to optimize a mathematical function that can map an input, e.g., an image, to a desired output prediction, such as a classification label or bounding box coordinates, by learning from training data. The most basic building block of this function is the neuron, which is based on the idea of the perceptron [Rosenblatt, 1958]. It maps an input vector x to an output $\hat{y}$ by multiplying it with the weight vector w, adding a bias b and then applying an activation function Ψ:

$$\hat{y} = \Psi\left(w^\top x + b\right). \tag{2.1}$$



(a) (b)

Figure 2.1: Neurons (a) form a Fully Connected (FC) Neural Network (b). The elements of the input vector x to the neuron get multiplied with the elements of the weight vector w. The results are added up together with the bias b and an activation function Ψ is applied, yielding the prediction $\hat{y}$. Multiple neurons can be combined into a FC Neural Network, with intermediate (hidden layers) and a final output layer.

Multiple neurons form a Fully Connected (FC) Neural Network when they are stacked in sequential layers and the output from each neuron of one layer forms the input of each neuron in the following layer (cf. Figure 2.1). This architecture, initially conceptualized by Ivakhnenko and Lapa [1966], can handle more complex functions. The output of all the neurons of a layer i is described by a vector Θ_i:

$$\Theta_i(x) = \Psi_i \left(W_i \Theta_{i-1}(x) + b_i \right), \tag{2.2}$$

with the weight matrix W_i, the bias vector b_i and the output of the previous layer $\Theta_{i-1}(x)$. Hence, the final output of a FC can be recursively defined by the intermediate layers, which are often referred to as hidden layers:

$$\hat{y} = \Psi_I \left(W_I \Psi_{I-1} \left(W_{I-1} \ldots \left(\Psi_1 \left(W_1 x + b_1 \right) \ldots \right) + b_{I-1} \right) + b_I \right), \tag{2.3}$$

where I is the total number of layers of the network. Using θ to denote the learnable parameters $\{W_i, b_i\}_{i=1}^{I}$ of the network, we can simply write this as

$$\hat{y}_n = \Theta_\theta(x_n) \tag{2.4}$$

for a given input x_n.

Please note that, when the activation function Ψ is linear, Equation (2.3) can

be simplified using matrix multiplication to a network with a single layer, nullifying the effect of using multiple layers. Therefore, non-linear activation functions, such as rectified linear unit (ReLU) [Nair and Hinton, 2010], leaky ReLU or Swish [Ramachandran et al., 2018] have to be employed.
In order to tune the function defined by the neural network (Equation (2.3)) for a given task, the learnable parameters need to be optimized. To this end, a loss function ($\mathcal{L}$) is defined that can measure how well the network performs with the current weights and biases. A typical choice is the $\mathcal{L}_2$ loss:

$$\mathcal{L}_2 = \sum_{n=1}^{N} (\hat{y}_n - y_n)^2, \tag{2.5}$$

where N is the number of samples in the dataset, y_n denotes the ground truth prediction for the n-th sample given by the training dataset and $\hat{y}_n$ is the network's prediction for the corresponding input x_n. Alternatively, the $\mathcal{L}_1$ function can be employed, replacing the L_2 with the L_1 norm, which is less sensitive to outliers in the data. Another important loss function is the binary cross-entropy loss [Shannon, 1948; Kullback and Leibler, 1951], which is often used for binary classification

$$\mathcal{L}_{\text{BCE}} = -\frac{1}{N} \sum_{n=1}^{N} \Big[y_n \log(\hat{y}_n) + (1 - y_n) \log(1 - \hat{y}_n) \Big]. \tag{2.6}$$

When generalized to multiple classes, it is called the categorical cross-entropy loss.
Once a suitable loss has been chosen, the network's parameters (weights and biases) can be adjusted to minimize that loss, maximizing performance for a given training dataset. This optimization is performed by computing the gradient of the loss function with respect to each parameter using backpropagation [Linnainmaa, 1970; Rumelhart et al., 1986]. Then, applying gradient descent, the parameters are updated repeatedly in the direction of the steepest descent for the average of all training samples:

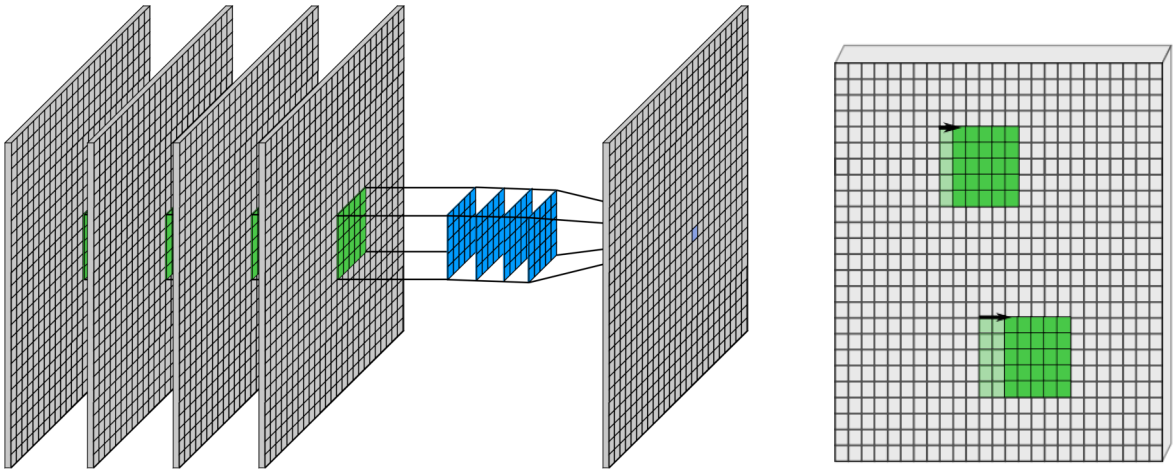
$$\theta \leftarrow \theta - \eta \nabla_\theta \mathcal{L}, \tag{2.7}$$

with the learning rate η controlling the step size. In practice, it is more computationally efficient to calculate the descent only for a subset of the training dataset for each step. This is called mini-batch gradient descent but is

also commonly referred to as stochastic gradient descent (SGD), even though SGD actually means the special case where the batch size is exactly one. This optimization strategy is further refined by algorithms such as AdaGrad [Duchi et al., 2010], RMSProp or Adam [Kingma and Ba, 2015] which try to speed up and stabilize learning.
Normalization is another common technique to improve the training process of neural networks [Ioffe and Szegedy, 2015; Ba et al., 2016; Ulyanov et al., 2017]. For example, in batch normalization the input to each layer across the mini-batch is normalized, which usually leads to faster convergence and improved model stability.

2.2 Convolutional Neural Networks

As we mainly work with image data, we make intensive use of Convolutional Neural Networks (CNNs) [LeCun et al., 1989b], a specialized neural network architecture that leverages reasonable assumptions about this domain. The first assumption is that the spatial proximity of pixels, i.e., input variables, in the two-dimensional structure of images indicates correlation and, thus, local features can be extracted that capture essential information in an image. CNNs take advantage of this by using relatively small kernels (filters) to extract such local features instead of considering all input elements at once, as a fully connected neural network would. The second assumption is that image features are invariant to translations. For example, extracting features for recognizing corners is assumed to be independent of the location of the corner in the image. Unlike fully connected neural networks, CNNs are inherently shift invariant, as the weights of each kernel remain the same when applied to different locations (parameter sharing). By incorporating this domain knowledge, CNNs need fewer parameters and are much less prone to overfitting than FC neural networks.
Each pixel of the d channels of an image is an input variable to the network. These are passed through convolutional layers that employ three-dimensional filters of size $k \times k \times d$, with each element representing a learnable parameter (weight), to calculate a feature map from the input. The weights of the kernel are multiplied with corresponding input variables of a window of the same



(a) (b)

Figure 2.2: Illustration of convolutions. (a) A $5 \times 5 \times 4$ kernel (blue) is applied to a $25 \times 25 \times 4$ input, generating a single feature map. The kernel consists of learned weights that operate on an equally sized region of the input (green) and produce a single output (light blue cell on the right feature map). (b) The kernel moves along the input and the calculation is repeated for different positions with the same weights. The number of pixels the kernel shifts at each step is called stride s, which is illustrated with the arrow for two examples (top: $s = 1$ and bottom: $s = 2$).

size. The results are added up and combined with the bias term, yielding a single element of the feature map. Thus, only local inputs are considered, leading to the extraction of local features as discussed above. The center of the kernel is moved along the input with a stride s and the procedure is repeated until the whole image is processed. Usually multiple kernels are applied to the input, determining the number of output feature maps. For deeper layers, this whole procedure, which is illustrated in Figure 2.2, is repeated with different filters on the intermediate outputs.

Apart from convolutional layers, traditional CNNs also include pooling or other suitable layers for down-sampling and FC layers for generating the final output vector. For instance, Figure 2.3 shows LeNet [LeCun et al., 1998], which demonstrates the typical architectural building blocks of early convolution-based image classification networks. Down-sampling layers decrease the spatial dimension of the input for subsequent convolutions, leading to a reduction of information and computational complexity. This can be achieved by employing convolutions with a stride larger than one (all convolutional networks [Springenberg et al., 2015]) or with pooling

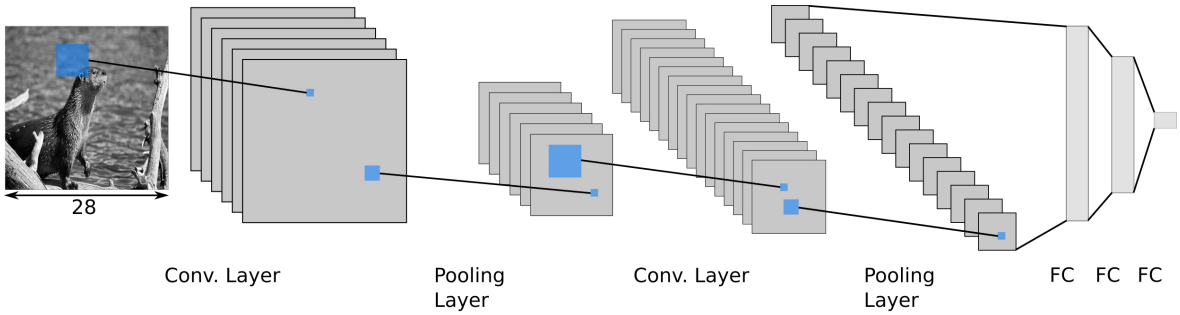


Figure 2.3: Illustration of LeNet based on [LeCun et al., 1998]. It shows the different layers of early CNNs. The convolutions compute multiple feature maps using convolutional kernels (blue). In between the convolutions, pooling layers are used to downsample the size of the feature maps, increasing the receptive field. Finally, the features are classified using several fully connected (FC) layers.

layers. LeNet employs average pooling [LeCun et al., 1989a], calculating the value of an element in the output feature map with the mean of all elements within a window of the input feature map. Finally, the FC layers map the features computed by the convolutions to a vector, which can be used, e.g., for classification. The design of LeNet was rapidly improved upon in the context of the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) [Russakovsky et al., 2015], requiring the classification of up to 1000 object classes on ImageNet [Deng et al., 2009].

In 2012, Krizhevsky et al. [2012] introduced AlexNet, a deep neural network with around 60 million parameters, outperforming its competitors by a large margin. It is based on the architecture of LeNet, but leverages five convolutional layers instead of only two, each with significantly more feature maps. Moreover, it employs max-pooling instead of average pooling, the ReLU activation function and dropout for regularization. The following year, Zeiler and Fergus [2014] improved upon AlexNet with ZFNet, visualizing feature maps to find a better architecture. They reduced the size of the initial filter from 11×11 to 7×7 and lowered the stride from $s = 4$ to $s = 2$, but the overall design remained largely unchanged (see Figure 2.4). In 2014, VGG [Simonyan and Zisserman, 2015] and Inception [Szegedy et al., 2014] further improved the classification performance on ImageNet with even deeper models. One of the challenges of this approach of adding more layers to the network is the vanishing or exploding gradients problem [Bengio

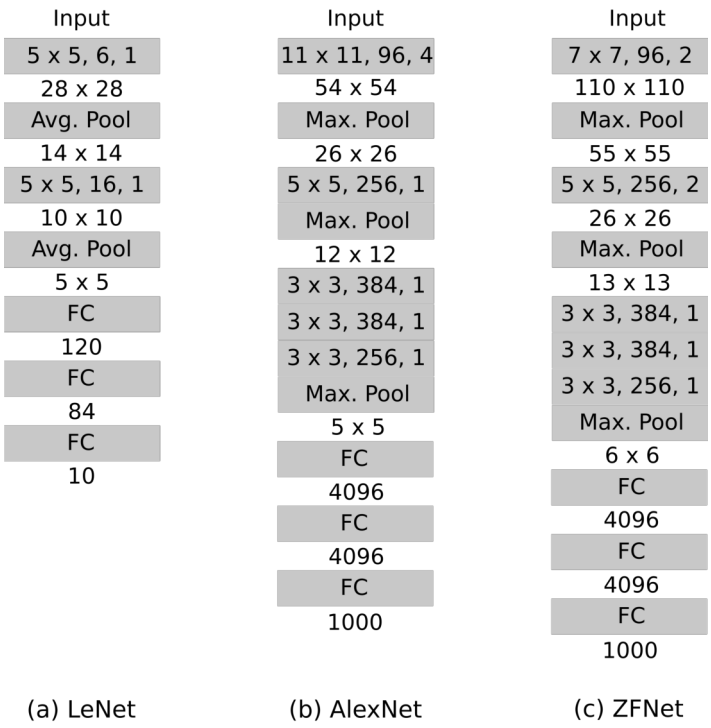


Figure 2.4: Comparison of LeNet [LeCun et al., 1998], AlexNet [Krizhevsky et al., 2012] and ZFNet [Zeiler and Fergus, 2014]. Each gray box represents a layer of the network. The convolutions are described by their kernel size (width × height), number of output feature maps and stride, in that order. Pooling layers are specified as either max-pooling or average pooling, while FC refers to fully connected layers. The size of the output is shown by the number below each layer, with width and height for two-dimensional outputs and vector length for the one-dimensional case.

et al., 1994; Glorot and Bengio, 2010]. With VGG, researchers at Oxford's *V*isual *G*eometry *G*roup overcame the vanishing gradient problem with weight initialization [Glorot and Bengio, 2010], building a network that is between 11 (VGG-11) and 19 (VGG-19) layers deep. Unlike others [Krizhevsky et al., 2012; Zeiler and Fergus, 2014; Szegedy et al., 2014], they used only 3×3 kernels, arguing that the receptive field of larger filters can be achieved with stacks of smaller kernels while needing fewer parameters (cf. Figure 2.5).
Szegedy et al. [2014], on the other hand, rely on "Inception modules", which apply 1×1, 3×3 and 5×5 convolutions to the input in parallel and concatenate the feature maps afterwards. Additionally, Inception applies 1×1 kernels prior to the compute-heavy larger filters to decrease the dimensionality following the network in network [Lin et al., 2013] structure to reduce the computational requirements. Later versions of the model continue using this design [Szegedy

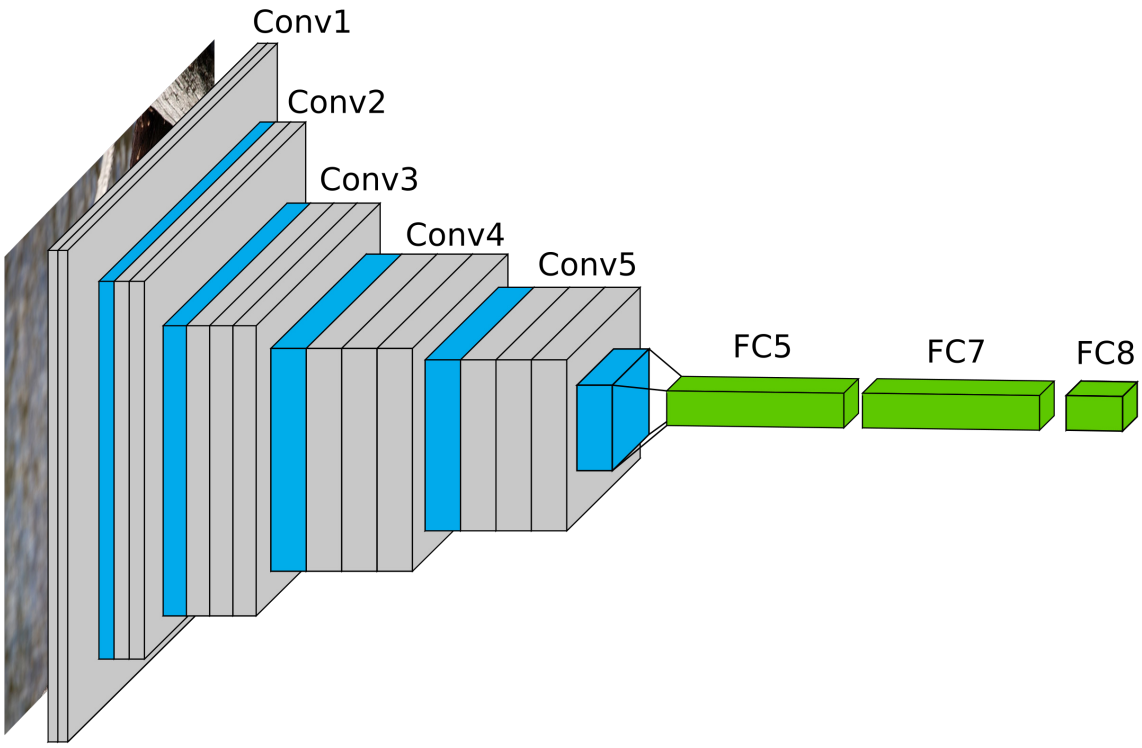


Figure 2.5: Illustration of VGG-16 [Simonyan and Zisserman, 2015]. It shows the structure of convolutional layers (gray) assembled as blocks (Conv1, Conv2, etc.) and followed by a pooling layer (blue). The final green layers illustrate fully connected (FC) layers.

et al., 2016, 2017].

Building upon the ideas of its predecessors, ResNet [He et al., 2016] follows the same basic structure as VGG, employing blocks of 3×3 kernels, and it includes bottleneck layers with 1×1 convolutions, similar to Inception. It employs an even deeper network, solving the performance degradation problem [He and Sun, 2014; Srivastava et al., 2015] with skip connections, adding the input to a stack of convolutional layers directly to the output of the stack.

The basic architectures introduced in this section have been adopted for a variety of tasks that are relevant to this thesis, such as measuring image quality with Inception [Heusel et al., 2017], computing perceptual loss with VGG [Johnson et al., 2016], performing face recognition with ZFNet, Inception or ResNet [Schroff et al., 2015; Deng et al., 2019] and object detection with ResNet-based backbones [Redmon and Farhadi, 2018; Jocher et al., 2023].

2.3 Object and Face Detection

For this thesis, object detection, specifically face detection, is employed as a part of the de-identification pipeline, localizing the areas to anonymize when no human-labeled ground truth is available. Moreover, training face detection models (i.e., YOLOv8 [Jocher et al., 2023] and DSFD [Li et al., 2019]) on anonymized data is one of our main methods to quantify data utility retention of de-identification approaches. Beyond its direct application in our research, it is important, for example, due to its use as part of face recognition pipelines [Schroff et al., 2015; Deng et al., 2019; Huang et al., 2020] or in healthcare [Davoudi et al., 2019; Liu et al., 2022; Selvaraju et al., 2022; Lee and Park, 2022]. Successful detection approaches need to be robust against complex real-world variations, including diverse poses or illumination conditions, occlusions and variance in object scale. Recent progress in this field has been largely driven by deep learning, significantly improving accuracy even for challenging conditions.

2.3.1 Traditional Handcrafted Features

Nevertheless, earlier approaches relying upon handcrafted features and traditional machine learning techniques, such as support vector machines (SVM) [Cortes and Vapnik, 1995], laid the foundations for the current success. Such features can be obtained, e.g., using Scale-Invariant Feature Transform (SIFT) [Lowe, 1999] or Histogram of Oriented Gradients (HOG) [Dalal and Triggs, 2005]. HOG computes the gradients for local image patches and accumulates their magnitudes into bins corresponding to their orientation. The resulting representation can be classified, for example, by an SVM to confirm the presence of an object within the regarded window. While later work like Deformable Part Models (DPM) [Felzenszwalb et al., 2010] further improved detection with handcrafted features, they were ultimately outperformed by deep learning-based approaches.

2.3.2 Deep Learning-Based Approaches

Deep learning-based approaches can be categorized into one-stage detectors and two-stage detectors. One of the earliest of these approaches to object

detection was the two-stage R-CNN [Girshick et al., 2013] (Regions with CNN features), which treats detection as two subtasks: First, it proposes around 2000 category-agnostic regions per image that are likely to contain an object using selective search [Uijlings et al., 2013]. In the second task, the network makes the actual bounding box and class label predictions. To this end, the region proposals are cut out, warped into a fixed-size image and passed as an input to AlexNet (cf. Section 2.2). The 4096-dimensional feature vectors calculated by the last convolutional layer for each image are classified using an SVM. Finally, a linear regressor model, which is inspired by DPM [Felzenszwalb et al., 2010], and non-maximum suppression (NMS) are applied to refine the bounding box predictions.
Even though later variants, such as Fast R-CNN [Girshick, 2015] or Faster R-CNN [Ren et al., 2015], improve upon the computational speed of R-CNN, the two-stage approach can be a bottleneck. Therefore, in order to achieve real-time detection, YOLO (You Only Look Once) [Redmon et al., 2015] formulates object detection as a single-stage regression problem. Unlike R-CNN, it processes the entire image at once, predicting all bounding boxes and class labels in a single network pass. To achieve this, YOLO uses a grid of S × S cells, each responsible for the prediction of B bounding boxes, their respective confidence scores ($\hat{K}$) and C class probabilities (p_c) (see Figure 2.6).

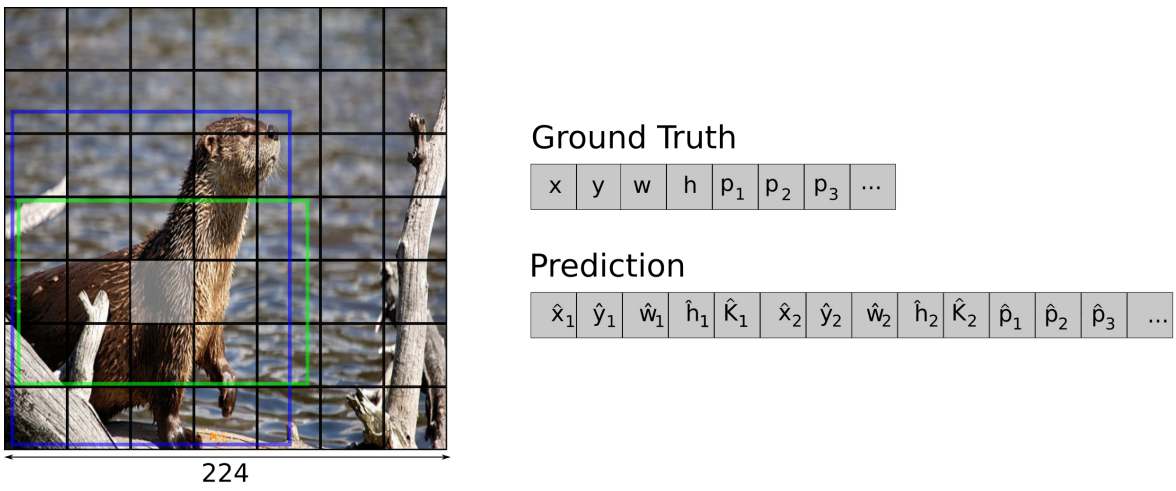


Figure 2.6: YOLO divides the input image into a grid of cells. For each cell, there is an associated ground truth and a prediction made by the model. In this example, the image is divided into 7 × 7 cells ($S = 7$) and two bounding boxes (blue and green) are predicted per cell ($B = 2$).

As each box is represented by four parameters, the x and y coordinates of its center and its width (w) and height (h), the total dimension of the output vector is:

$$\dim_{\text{YOLOv1}} = S \times S \times ((4 + 1) \times B + C). \tag{2.8}$$

For example, when applying YOLO to the PASCAL VOC dataset [Everingham et al., 2008], the number of ground truth classes is 20 (C = 20) and the authors set S = 7 and B = 2, resulting in a 1470-dimensional output.
To compute these predictions, the detector employs an Inception-based network architecture (cf. Section 2.2). Initially, a smaller version of the model consisting of only 20 convolutional layers and a single fully connected layer is pretrained for classification on ImageNet [Deng et al., 2009] with an input resolution of 224×224. Then, to adapt the model for detection, the resolution is doubled to 448×448, four additional convolutional layers are appended and the fully connected layer is exchanged for two new ones.

2.3.3 Object Detection with YOLOv8

Even though YOLO achieves remarkable results when compared to other real-time detectors developed at the same time, it is less accurate than the slower two-stage approaches. Later versions address this issue while maintaining or even improving processing speed. Here, we discuss YOLOv8 [Jocher et al., 2023], the version we use for our main experiments in Section 4.3.
The network architecture of YOLOv8 can be divided into three parts: a backbone, a neck and a head (see Figure 2.7). The objective of the backbone is to extract features from the image that are useful for object detection. YOLOv8 employs a Darknet-based backbone [Redmon and Farhadi, 2018] with cross-stage partial connections (CSP) [Wang et al., 2019], which significantly reduce the necessary computations. The neck connects the backbone to the head, refining the semantic and spatial representation of features before a prediction is made. In YOLOv8 it consists of Spatial Pyramid Pooling Fast (SPPF) [He et al., 2014; Jocher, 2020] and a PANet-like block [Liu et al., 2018]. The head is divided into three parts, each predicting the bounding boxes, objectness scores, and class probabilities for different spatial scales. Inspired by YOLOX [Ge et al., 2021], it employs separate branches for classification and bounding box prediction. This structure results in better performance as it reduces task

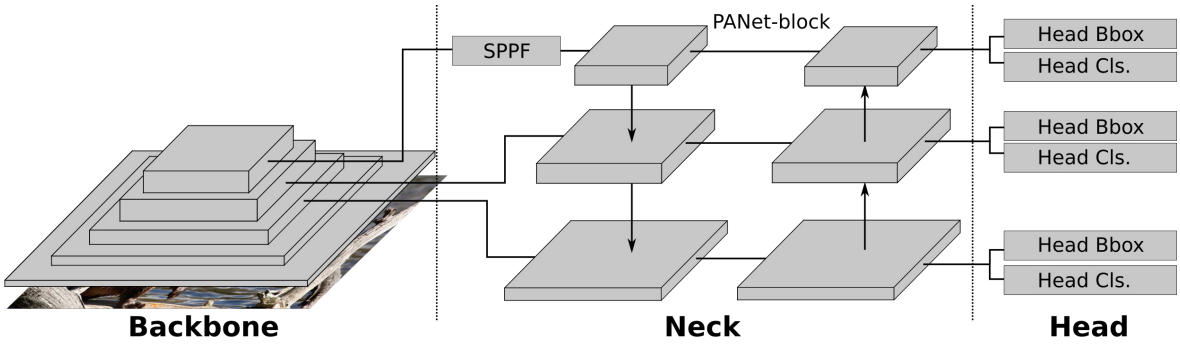


Figure 2.7: Overview of the YOLOv8 architecture. The network can be divided into three parts: a backbone for basic feature extraction, a neck for refining the semantic and spatial representation of features and a head for making the predictions at three different scales with independent branches for classification (cls.) and bounding box prediction (Bbox).

conflicts.
A key characteristic of the YOLOv8 design is its scalability, allowing the user to select a model size that balances computational efficiency and accuracy according to the requirements. It has five variants: YOLOv8n (nano), YOLOv8s (small), YOLOv8m (medium), YOLOv8l (large) and YOLOv8x (extra-large), each progressively increasing both the number of convolutional layers (depth) and the number of feature channels per layer (width).
Like the architecture, YOLO's loss function has evolved with the different versions. In YOLOv8, Complete Intersection over Union (CIoU) loss [Zheng et al., 2020] and Distribution Focal Loss (DFL) loss [Li et al., 2020] are used to train bounding box prediction and cross-entropy loss (cf. Section 2.1) for classification.
Overall, these improvements have helped YOLOv8 find a good balance between accuracy and computational demands and established it as an approach for many real-world applications such as medical object detection [Ragab et al., 2024] or face detection [Qi et al., 2021].

2.3.4 Face Detection with DSFD

While general object detection models like YOLOv8 can be trained for face detection, there are domain-specific challenges, such as extreme variations in object size. Therefore, specialized face detectors have been developed [Zhang et al., 2016; Li et al., 2019; Liu and Tang, 2020; Deng et al., 2020; Liu et al.,

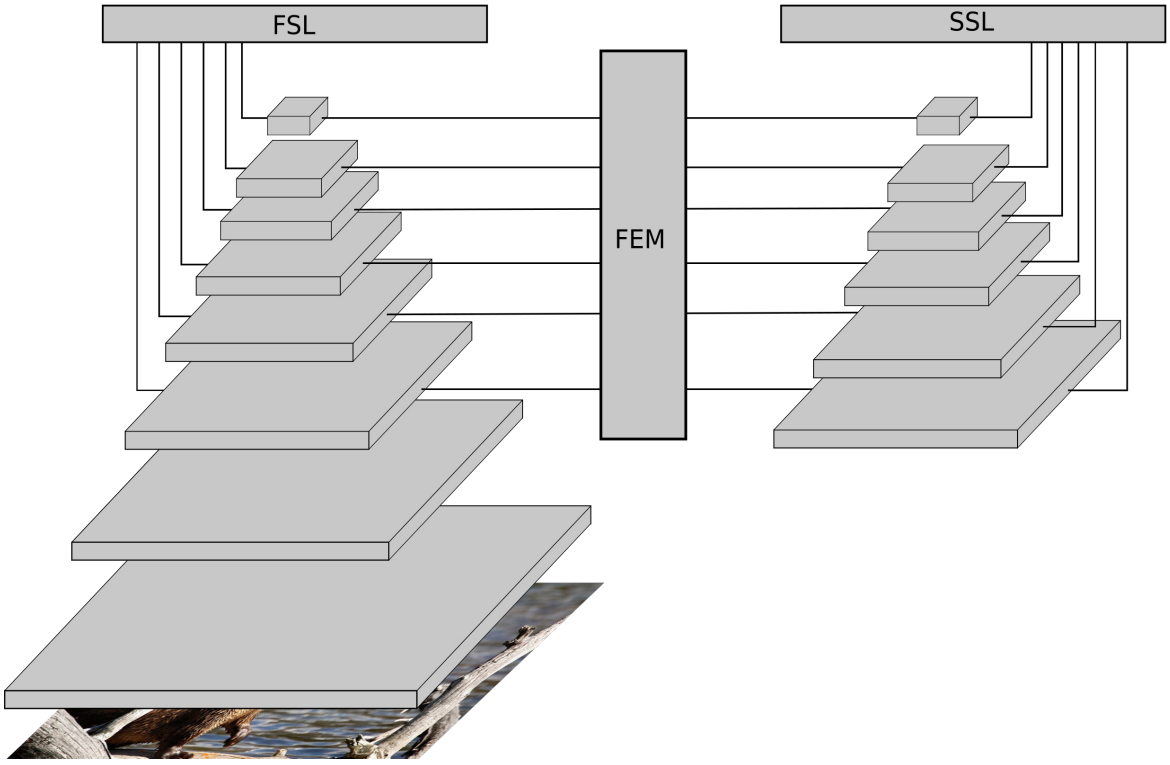


Figure 2.8: Overview of the Dual Shot Face Detector (DSFD) architecture. The Feature Enhance Module (FEM) and a two-term loss consisting of first shot loss (FSL) and second shot loss (SSL) improve face detection on widely varying scales.

2020; Zhang et al., 2021; Guo et al., 2022], of which we use Dual Shot Face Detector (DSFD) [Li et al., 2019] as detailed in Section 4.3. DSFD improves face detection across scales by adopting a two-stream design (see Figure 2.8). It relies on a modified fully convolutional VGG-16 backbone (cf. Section 2.2) with additional blocks of convolution layers to make the first shot predictions. To this end, it extracts the features from six different stages of the network and passes them through an SSD-like [Liu et al., 2015a] head. It employs the Feature Enhance Module (FEM) for better feature integration and makes the second shot predictions from the enhanced feature maps. The relation between the cells of the enhanced feature map (ec) and those from the original (oc) is defined by:

$$\mathrm{ec}_{(i,j,l)} = f_{\mathrm{concat}}(f_{dilation}(\mathrm{nc}_{(i,j,l)})) \tag{2.9}$$

$$\mathrm{nc}_{(i,j,l)} = f_{\mathrm{prod}}(\mathrm{oc}_{(i,j,l)}, f_{\mathrm{up}}(\mathrm{oc}_{(i,j,l+1)})), \tag{2.10}$$

with the non-local neuron cells nc, the indices i,j denoting the cell location within the feature map in layer l, and f describing the application of concatenation, dilation (atrous convolution [Chen et al., 2016]), product, or up-sampling. This architecture is combined with progressive anchor loss (PAL), which assumes different anchor sizes for the first and second shot predictions, using two separate terms, first shot loss (FSL) and second shot loss (SSL), to increase detection accuracy. Additionally, DSFD uses improved anchor matching for better initialization of the regressor.

2.4 Face Recognition

While face detection, as discussed in the last section, means the localization of faces within an image, face recognition means the unique identification of a specific person based on facial features. In this thesis, we employ two face recognition tools, FaceNet [Schroff et al., 2015] and ArcFace [Deng et al., 2019], to evaluate the privacy protection of anonymization approaches. Here, we first give an overview of this technology, focusing on research that directly influenced the development of the tools we use and later we briefly discuss how the human visual system identifies faces.

2.4.1 Machine Learning-Based Face Recognition

An early approach to automated face recognition has been Eigenfaces [Turk and Pentland, 1991], which employs principal component analysis (PCA) to find eigenvectors (eigenfaces) for a set of faces. Then, a new face can be identified by comparing the coefficients of its eigenvector representation to those of the known faces. Later, researchers improved recognition using local handcrafted descriptors, such as local binary patterns [Ahonen et al., 2006; Hadid, 2008] and Gabor features [Liu and Wechsler, 2002; Zhang et al., 2005], or learning-based local descriptors [Cao et al., 2010], which offer greater robustness in uncontrolled conditions.
A significant breakthrough came with the application of deep learning, with approaches such as DeepFace [Taigman et al., 2014] rivaling human-level performance. However, this model treats recognition as a classification problem during training, using softmax loss to learn to assign faces to a closed set

of identities. To generalize recognition to new individuals, it has to rely on the intermediate representation being discriminative enough to differentiate it from existing embeddings, without explicit training for this.
In contrast, subsequent approaches switched towards loss functions inspired by metric learning, directly optimizing the model to produce a representation where distance reflects face similarity. For instance, DeepID2 relies on contrastive loss [Sun et al., 2014a,b, 2015] and FaceNet introduced triplet loss, which significantly improved recognition.
FaceNet. An overview of FaceNet is given in Figure 2.9. Inspired by Weinberger and Saul [2005], triplet loss directly enforces a small squared distance of embeddings f(x) of the same identity for intra-class compactness and, conversely, a large distance between different faces for inter-class separability. To achieve this, it uses image triplets consisting of an anchor image, a positive image, which has the same identity, and a negative image showing another

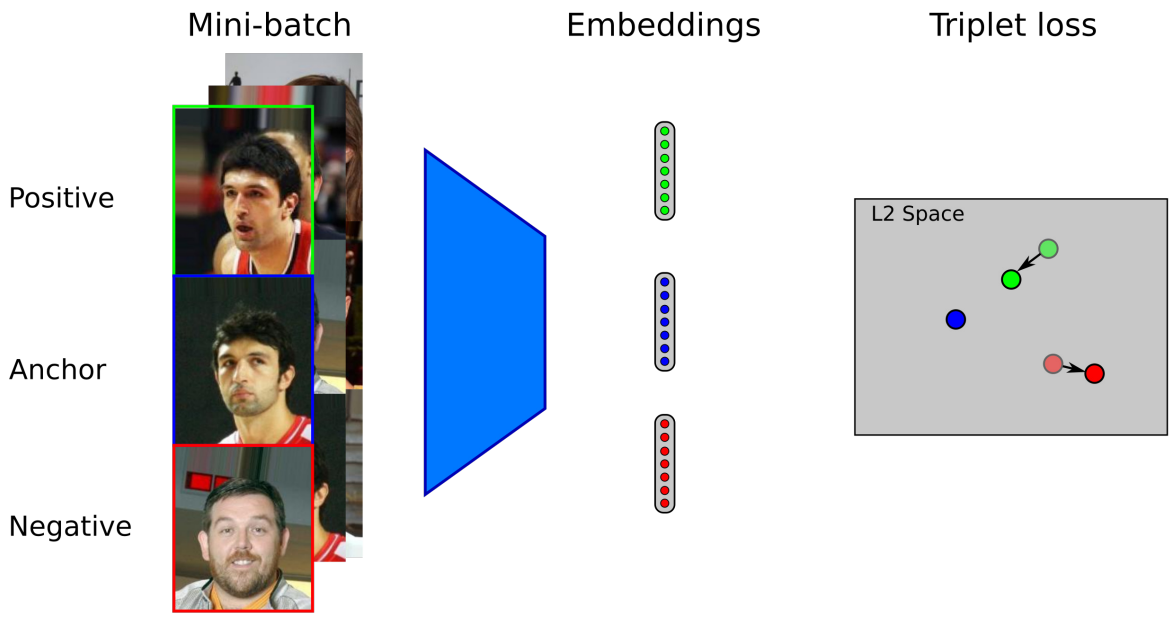


Figure 2.9: Overview of FaceNet. Each mini-batch contains anchor images, positive images of the same identity as the anchor and negative images showing a different person. These are passed through a ZFNet or Inception-based neural network (cf. Section 2.2) to create 128-dimensional embeddings. Afterwards, these are L_2 normalized and triplet loss is applied to directly enforce inter-class separability and intra-class compactness.

identity. Formally, the loss function can be expressed as:

$$\mathcal{L}_{\text{triplet}} = \sum_{i=1}^{N} \left[\|f(x_i^a) - f(x_i^p)\|_2^2 - \|f(x_i^a) - f(x_i^n)\|_2^2 + \delta \right]_+, \tag{2.11}$$

with N being the total number of all triplets within the set, x_i^a denoting an anchor image, x_i^p a positive image and x_i^n a negative image, while δ represents the margin enforced between positive and negative pairs.
Carefully selecting the right triplets during training is essential for FaceNet, as triplets that are too easy do not provide meaningful gradients, slowing down training. In theory, they should be chosen such that the distance of the positive sample to the anchor is maximized

$$\underset{x_i^p}{\text{argmax}} \, \|f(x_i^a) - f(x_i^p)\|_2^2, \tag{2.12}$$

and the distance between the anchor and the negative image is minimized

$$\underset{x_i^n}{\text{argmin}} \, \|f(x_i^a) - f(x_i^n)\|_2^2. \tag{2.13}$$

In practice, finding these hard samples globally is computationally inefficient and the authors of FaceNet instead employ a novel online triplet mining method, choosing only from the images of each mini-batch. To ensure enough positive images of each individual are included, approximately 1,800 examples were sampled for each iteration with 40 faces of each identity and additional random negative faces. All anchor-positive pairs were used without mining, while anchor-negative pairs were selected using semi-hard exemplars that satisfy the condition

$$\|f(x_i^a) - f(x_i^p)\|_2^2 < \|f(x_i^a) - f(x_i^n)\|_2^2, \tag{2.14}$$

enforcing that they are farther from the anchor embedding than the positive image. This avoids getting trapped in bad local minima during early training. To create the embeddings f(x) on which the loss function operates, FaceNet relies on standard network architectures, either Inception or ZFNet (cf. Section 2.2), modified with 1×1 convolutions for dimensionality reduction [Lin et al., 2013]. These 128-dimensional vectors are L_2 normalized ($\|f(x)\|_2 = 1$) and form a face similarity embedding space that generalizes well to new,

unseen faces and can perform face verification, identification or clustering. Despite FaceNet's success, the computational cost associated with triplet mining is a significant downside. Therefore, later research has increasingly adopted softmax loss again and combined it with a margin penalty to encourage inter-class separability and intra-class compactness. Liu et al. [2016] introduce L-Softmax (large-margin softmax), which learns from the entire mini-batch without triplet selection. Building upon this idea, Liu et al. [2017] developed SphereFace, which uses A-Softmax (angular softmax), constraining the learned representation of faces to be discriminative on a hypersphere. They employ an angular decision boundary, leveraging the intrinsically angular distribution of features learned with softmax, and enforce an angular decision margin.

ArcFace. Deng et al. [2019] follow the basic structure of these softmax-based face recognition systems (see Figure 2.10), but improve upon the loss function by introducing Additive Angular Margin Loss (ArcFace), which directly corresponds to geodesic distance on a hypersphere and further increases the discriminative power of the learned face representations. Here, we explain ArcFace following the reasoning given by the authors, starting with the softmax loss. It can be derived from categorical cross-entropy loss (see Section 2.1) by

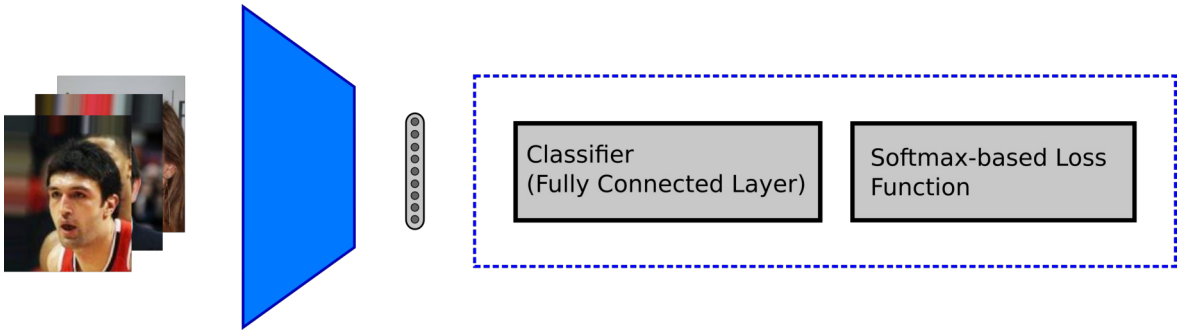


Figure 2.10: Overview of typical softmax-based face recognition systems, such as **ArcFace**. Images are processed by a classifier-based network (ResNet-50, ResNet-100) to create face embeddings. During the training phase, the embeddings are passed through a fully connected layer for classification and a softmax-based loss function (e.g., ArcFace) is applied. During test time, the elements connected to softmax loss are removed (blue dashed box) and recognition is performed by directly calculating the distance between existing embeddings and the test face.

inserting the softmax function for the network's prediction $\hat{y}$:

$$\mathcal{L}_{\text{softmax}} = -\frac{1}{N}\sum_{i=1}^{N}\log\frac{e^{W_{y_i}^T r_i + b_{y_i}}}{\sum_{c=1}^{C} e^{W_c^T r_i + b_c}}, \tag{2.15}$$

with $r_i = f(x_i)$ denoting the embedding of a face image (x_i) and y_i being the associated ground truth class label. C is the total number of classes, W_c is the c-th column of the weight matrix W of the fully connected layer used for classification during training and b is the bias. Following Liu et al. [2017], the bias of the classification layer is set to zero, as Liu et al. [2016] showed that this does not affect performance and simplifies the equation. Thus, the prediction can be expressed depending on the angle ϕ_c between the weights W_c and the embedding r_i:

$$\hat{y} = W_c^T r_i = \|W_c\|_2 \|r_i\|_2 \cos(\phi_c). \tag{2.16}$$

Inspired by previous work [Wang et al., 2017, 2018a] L_2 normalization is applied to the weights as well as to the embeddings and the latter are scaled to κ, constraining them to a hypersphere:

$$\|r_i\|_2 = \kappa, \quad \|W_c\|_2 = 1 \quad \forall i, c. \tag{2.17}$$

Combining this with Equation (2.15) results in a modified expression for the softmax loss, which depends on ϕ and, thus, allows for using an angular decision boundary. We finally arrive at the ArcFace loss by applying an additive angular margin penalty m

$$\mathcal{L}_{\text{ArcFace}} = -\frac{1}{N}\sum_{i=1}^{N}\log\frac{e^{\kappa\cdot\cos(\phi_{y_i}+m)}}{e^{\kappa\cdot\cos(\phi_{y_i}+m)} + \sum_{c\neq y_i}^{C} e^{\kappa\cdot\cos(\phi_c)}}, \tag{2.18}$$

which enforces inter-class separability and intra-class compactness.

2.4.2 Face Recognition by Humans

In addition to automated computer-based face recognition, we want to briefly discuss how the human visual system perceives and identifies faces. We will use the obtained insights in Sections 4.2 and 4.3 as a basis when visually assessing the extent to which de-identification approaches alter an individual's appearance.

First, while humans rely on holistic processing for recognition [Young et al., 1987], faces can often be identified from a single feature [Davies et al., 1977]. Experimental results suggest that the most important features for identification are the eyes, followed by the mouth and nose [Sinha et al., 2006]. Additionally, according to a study by Sadr et al. [2003], the eyebrows are also crucial. Abudarham and Yovel [2014] attempt to pinpoint the decisive features influencing the human recognition system, constructing an explicit face space from 20 facial attributes: lip thickness, hair color, eye color, eye shape, eyebrow thickness, ear protrusion, forehead height, hair length, eye size, skin texture, jaw width, eyebrow shape, nose size, nose shape, skin color, face proportions, cheek shape, eye distance and mouth size. Their findings suggest that features connected with a high perceptual sensitivity (PS) like eyebrow thickness, hair color, eye shape, eye color and lip thickness, are more important for recognition by humans than those with low PS (skin color, face proportion, eye distance, nose and mouth size). They argue that the former stay constant across multiple appearances of the same individual and are, therefore, suitable for learning identities.

2.5 Deep Learning-Based Image Generation

Apart from discriminative tasks like object detection or face recognition, deep learning can also be employed for generative tasks, such as synthesizing images or other data. Recently, this has been applied for image super-resolution [Ledig et al., 2017; Wang et al., 2018c, 2021b], inpainting [Hukkelås et al., 2020; Yeh et al., 2017], text-to-image generation [Reed et al., 2016; Qiao et al., 2019; Tan et al., 2023] or face swapping [Nirkin et al., 2019, 2022; Chen et al., 2020a]. In the context of this thesis, Generative Adversarial Networks (GANs) [Goodfellow et al., 2014] and diffusion models [Sohl-Dickstein et al., 2015; Ho et al., 2020] form an essential part of the de-identification approaches discussed in Sections 3.2.3 and 4.1. Thus, we give an overview of the underlying ideas, with an emphasis on the methods we use directly in our work: StyleGAN2 [Karras et al., 2020], FSGANv2 [Nirkin et al., 2022] and Stable Diffusion [Rombach et al., 2022].

2.5.1 Generative Adversarial Networks

GANs leverage the concept of adversarial learning, with the generator (G) and discriminator (D), both typically implemented as neural networks, acting as adversaries. While the generator tries to map a latent vector z from the distribution p_z to a synthetic output resembling a sample from p_{data}, the discriminator attempts to classify these outputs as belonging to the real training dataset or being synthetically generated (fake).
The intuition behind this is that the generator can learn to improve its output with the feedback from the discriminator, while at the same time, the discriminator is forced to improve to detect the increasingly realistic fake samples produced by the generator. Thus, in the end, the generator is able to produce outputs that cannot be distinguished from samples drawn from the real distribution p_{data}.
For a more formal description, following Goodfellow et al. [2014], the objective function of the discriminator in this process can be expressed using cross-entropy loss (compare Equation (2.6)) as:

$$\max_D \mathcal{L}_{\text{adv}}(D) = \mathbb{E}_{x\sim p_{\text{data}}}[\log D(x)] + \mathbb{E}_{z\sim p_z}[\log(1 - D(G(z)))]. \tag{2.19}$$

The loss of the generator is:

$$\min_G \mathcal{L}_{\text{adv}}(G) = \mathbb{E}_{z\sim p_z}[\log(1 - D(G(z)))]. \tag{2.20}$$

These two equations are commonly summarized with the following expression defining the minimax game between the generator and the discriminator:

$$\min_G \max_D \mathcal{L}_{\text{adv}}(G, D) = \min_G \max_D \mathbb{E}_{x\sim p_{\text{data}}}[\log D(x)] + \mathbb{E}_{z\sim p_z}[\log(1 - D(G(z)))]. \tag{2.21}$$

Generator and discriminator are optimized alternately, keeping the weights of G constant when updating D and vice versa, until the generator produces the required quality. A description of this process is presented in Algorithm 1.
As shown by Goodfellow et al. [2014], optimizing the generator this way corresponds to minimizing the Jensen-Shannon divergence between $G(z)$ and p_{data}, resulting in the two distributions being indistinguishable if the training converges.

Algorithm 1 Training of GANs based on [Goodfellow et al., 2014].

for number of training iterations **do**

Sample M noise samples $z_{(i)}$ from the noise distribution p_z.

Sample M examples $x_{(i)}$ from the training dataset.

Update the discriminator weights (θ_D) by:

$$\nabla_{\theta_D} \frac{1}{m} \sum_{i=1}^{M} \left[\log D\left(x_{(i)}\right) + \log\left(1 - D\left(G\left(z_{(i)}\right)\right)\right)\right]$$

Sample M noise samples z from p_z.

Update the generator weights (θ_G) by:

$$\nabla_{\theta_G} \frac{1}{M} \sum_{i=1}^{M} \log\left(1 - D\left(G\left(z_{(i)}\right)\right)\right)$$

end for

Improving Loss Functions. Despite this promising theory, in practice, GANs often suffer from mode collapse and training instability. One of the causes of this is the use of binary cross-entropy loss for the generator, which is designed for binary classification, approaching zero when the discriminator can perform its objective with high confidence. This leaves the generator with vanishing gradients, unable to improve further. To address this issue, various alternatives to the original adversarial loss function have been suggested. Goodfellow et al. [2014] proposed to mitigate the problem by using Non-Saturating Loss, exchanging the generator loss with:

$$\mathcal{L}_{\text{adv}}(G) = -\frac{1}{2}\mathbb{E}_z[\log(D(G(z)))]. \tag{2.22}$$

This version of the loss was heuristically chosen to ensure a strong gradient. More recent formulations of the adversarial loss include least-squares GAN (LSGAN) [Mao et al., 2017], Wasserstein GAN (WGAN) [Arjovsky et al., 2017] or Wasserstein GAN with Gradient Penalty (WGAN-GP) [Gulrajani et al., 2017]. However, according to a study by Lucic et al. [2018], their differences could arise from different computational budgets and, therefore, more extensive hyperparameter optimization instead of the fundamental superiority of one algorithm over the other.

In addition to these changes in the general adversarial loss, others proposed application-oriented loss functions $\mathcal{L}_{\text{app}}$, which are typically combined with

the adversarial loss in the following manner [Pan et al., 2020]:

$$\mathcal{L}(G) = \lambda_{\text{app}}\mathcal{L}_{\text{app}} + \lambda_{\text{adv}}\mathcal{L}_{\text{adv}}, \tag{2.23}$$

where λ_{app} and λ_{adv} are weighting factors. An example is the pixel-wise loss, which can be implemented using an L_1 or L_2 norm-based loss (see Section 2.1). While this loss has helped to improve image quality for many applications [Isola et al., 2017; Nirkin et al., 2019; Pathak et al., 2016], it does not take into account higher-level artifacts visible by human perception. For this purpose, perceptual loss [Johnson et al., 2016] can be employed, which processes both the real and the generated image with a pretrained classification network, e.g., VGG-19, and compares several of their intermediate layer activations. The loss function is:

$$\mathcal{L}_{\text{perc}}(x, y) = \frac{1}{d_i w_i h_i} \sum_{i=1}^{I} \|(f_i(x) - f_i(y)\|, \tag{2.24}$$

where $f_i(x)$ corresponds to the activations of layer i for the input x and d_i, w_i and h_i are the dimensions of the resulting feature map.

Model Architectures. Apart from the loss, significant research has been done to improve GANs with new network architectures. While Goodfellow et al. [2014]'s original model was based on fully connected layers, Radford et al. [2016] developed an all convolutional architecture (DCGAN), improving training stability. However, the resulting images had a relatively low resolution of 64×64 pixels. Later, Karras et al. [2018] were able to generate high-quality images with 1024×1024 pixels by further stabilizing and speeding up the training. They achieved this by using a novel training methodology called progressive growing, which starts with a relatively shallow network creating low-resolution images, then progressively adding layers to the network and increasing the resolution. The new layers were faded in by initially treating them like residual blocks [He et al., 2016], multiplying their output with a weighting factor λ_{pro} and adding it to that of the previous layer weighted by $1 - \lambda_{pro}$. During training, λ_{pro} is increased linearly from 0 to 1, at which point the newly added layer behaves like any other.

Building on this methodology, StyleGAN [Karras et al., 2021b] further improved the state of the art concerning generated image quality. One of the key novelties has been the usage of an intermediate latent space W. Instead

of directly relying on the latent code sampled from p_z to initiate image generation, they first processed it through an eight-layer fully connected neural network called the mapping network to create W. The advantage of this is that, unlike the traditional input latent space, it is not forced to represent the statistics of the training dataset. This way, the mapping network is free to learn to transform the latent code to a disentangled representation during the usual GAN training. This can be done without specific supervision, as this is a more efficient representation, helping the generator to fool the discriminator. A learned affine transformation is applied to the vector from the intermediate latent code and the vector is passed to the synthesis network via adaptive instance normalization (AdaIN) [Huang and Belongie, 2017], which is often used in the style transfer literature. Additionally, the generator receives stochastic noise modified by a learned per-channel scaling factor

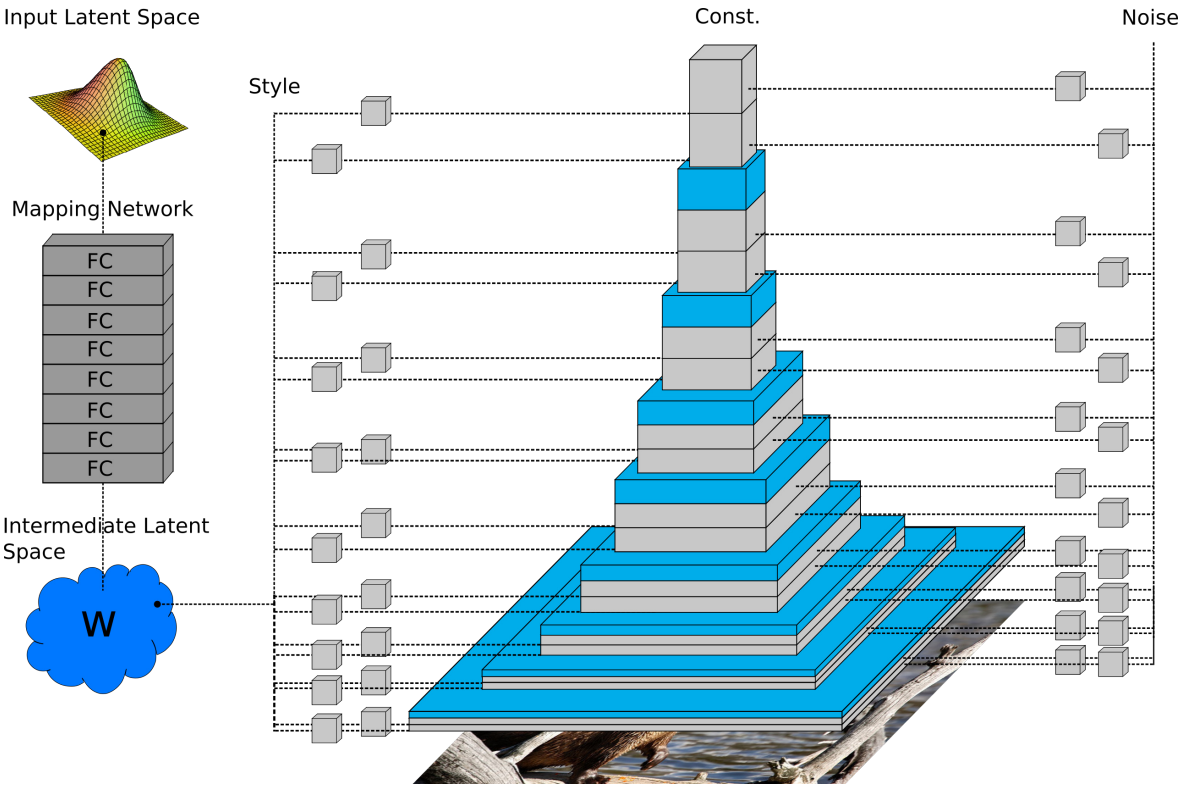


Figure 2.11: StyleGAN overview. The mapping network transforms a vector from the input latent space to the intermediate latent space, a more efficient disentangled representation. From there, a learned affine transformation is applied to the vector and it is passed to each of the layers of the synthesis network through AdaIN. The synthesis network uses this input together with random noise multiplied by a learned scaling factor to create 1024×1024 images conditioned by the style defined by the input vector.

as an input, which helps it to create stochastic properties of an image, like the placement of hairs and skin pores when generating faces. Without this component, the network would have to learn how to create spatially varying pseudorandom numbers, wasting network capacity. An overview of StyleGAN is given in Figure 2.11. For StyleGAN2, Karras et al. [2020] reworked parts of the architecture that caused artifacts in the generated images. This included changing normalization to avoid the destruction of information about the relative magnitude of features. Additionally, they replaced progressive growing with a modified version of MSG-GAN [Karnewar and Wang, 2020], as progressive growing causes an excessive focus on high-frequency details of the intermediate layers. Moreover, path length regularization has been introduced to encourage the network to construct an intermediate latent space with smoothly changing output images when interpolating along a path in the intermediate latent space, leading to a more consistent behavior of the model.

Conditioning. Another direction of research has focused on conditioning GANs to generate specific outputs. Mirza and Osindero [2014] developed conditional GAN (cGAN) which works by feeding the additional information c to both the generator and the discriminator through an extra input layer. The objective function from Equation (2.21) changes to:

$$\min_G \max_D \mathcal{L}_{\text{adv}}(D, G) = \mathbb{E}_{x \sim p_{\text{data}}(x)}[\log D(x|c)] + \mathbb{E}_{z \sim p_z(z)}[\log(1 - D(G(z|c)))]. \quad (2.25)$$

This simple conditioning is, however, not enough for many use cases, such as face swapping [Nirkin et al., 2019; Chen et al., 2020a] or face de-identification with data utility retention [Maximov et al., 2020; Hukkelås et al., 2019; Hukkelås and Lindseth, 2023; Leibl et al., 2023], which typically require conditioning by an input image. A popular approach for this is to use a U-Net [Ronneberger et al., 2015] based generator, which was pioneered by Isola et al. [2017] (see Figure 2.12). Their work combines $\mathcal{L}_1$ loss and adversarial loss (see Equations (2.21) and (2.23)) to generate realistic images from inputs, such as edge maps or semantic segmentations. Wang et al. [2018b] further improved upon this idea using a coarse-to-fine generator, a multi-scale discriminator, and by adding perceptual loss-based terms to the overall loss function.

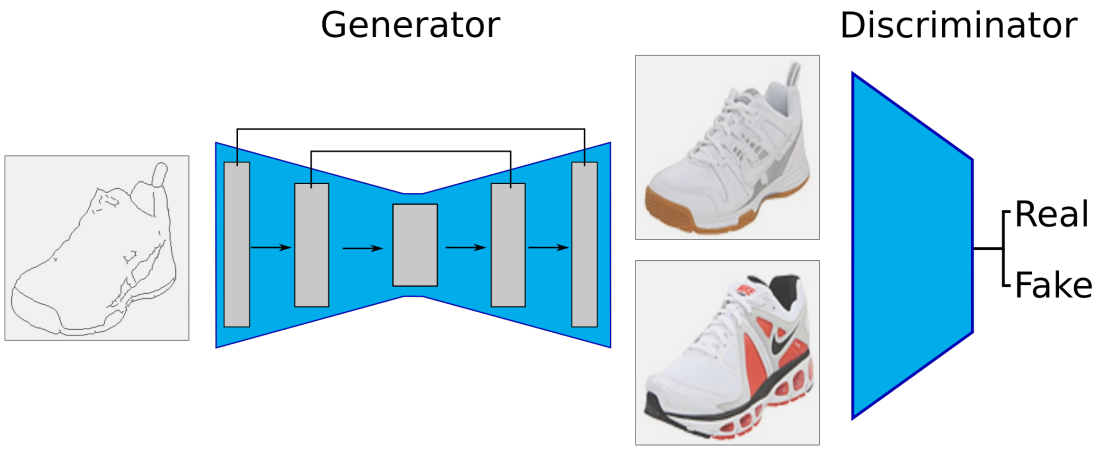


Figure 2.12: U-Net-based GAN. The U-Net utilizes the input image to condition the generator. The discriminator classifies the output as real or fake with the help of the corresponding ground truth image provided by the dataset.

Face Swapping with FSGANv2. One application of GANs, which is especially relevant for this thesis, is face swapping [Natsume et al., 2018b; Pumarola et al., 2018; Natsume et al., 2018a], replacing a target's face with that of a source. Among the most widely used early frameworks was DeepFaceLab, which was open-sourced in 2018 [1], with academic papers published only later [Petrov et al., 2020; Liu et al., 2023]. Another prominent method is FSGANv2 (Face Swapping GAN version 2) [Nirkin et al., 2019, 2022], which does not require subject-specific training, handles minor occlusions effectively, and produces highly realistic results. FSGANv2 consists of three main steps: reenactment and segmentation, face inpainting, and blending into the background.

In the first step, a segmentation mask of the source is generated and the recurrent reenactment generator (G_r) is used to transfer the position and expression of the source to the target face. G_r is trained as a GAN using an application-specific loss (see Equation (2.23)). The $\mathcal{L}_{\text{app}}$ term of the loss is a combination of $\mathcal{L}_1$ loss (cf. Section 2.1) and a domain-specific perceptual loss (compare Equation (2.24)) with the pretrained classifier (VGG-19) trained on face recognition and attribute classification datasets. The generator is conditioned with 98 landmarks extracted with the method discussed by Wang et al. [2021a]. As

[1] https://github.com/iperov/DeepFaceLab

the quality of the generated images suffers if the pose difference between the target and the source is too large, the reenactment generator works iteratively. To this end, the authors developed an additional model consisting of 12 fully connected layers, which estimates the landmarks in an intermediate position between the target's and the source's pose. The landmarks are then used as an input to G_r and the generation process is repeated until reaching the target pose.
In the second step, the generated face is compared to the segmentation mask of the source. The regions of the face that are not visible in the synthesized face due to occlusions in the original image are inpainted using the face inpainting generator (G_c).
Finally, in the third step, the generated face is adjusted to the skin tone and lighting conditions with a blending generator (G_b) trained with the Poisson blending loss proposed by the authors.

2.5.2 Diffusion Models

Recently, diffusion models [Sohl-Dickstein et al., 2015; Ho et al., 2020] have been shown to *beat* GANs on image generation in terms of quality and diversity [Dhariwal and Nichol, 2021]. The essential idea behind diffusion is to train a model to reverse the gradual addition of noise to images. The training cycle is divided into the forward and reverse process. During the forward process, Gaussian noise is progressively added to a training sample x_0 at each timestep t according to a variance scheduler β_t. Following Ho et al. [2020], this can be written as a Markov Chain of forward transitions:

$$q(x_{1:T}|x_0) := \prod_{t=1}^{T} q(x_t|x_{t-1}), \tag{2.26}$$

$$q(x_t|x_{t-1}) := \mathcal{N}(x_t;\ \sqrt{1-\beta_t}x_{t-1}, \beta_t\mathbf{I}). \tag{2.27}$$

Here, $\mathcal{N}$ denotes a normal distribution with mean $\sqrt{1-\beta_t}x_{t-1}$ and variance $\beta_t\mathbf{I}$. They also provide an equation to efficiently sample noisy images at an arbitrary timestep t directly from the initial image:

$$q(x_t|x_0) = \sqrt{\bar{\alpha}_t}x_0 + \sqrt{1-\bar{\alpha}_t}\epsilon, \tag{2.28}$$

with ϵ indicating sampling noise from the isotropic standard normal distribution ($\epsilon \sim \mathcal{N}(0, \mathbf{I})$) and $\bar{\alpha}_t = \prod_{s=1}^{t} \alpha_s$ with $\alpha_t = 1 - \beta_t$.
During the backward process, the denoising network, defined by its parameters θ, tries to predict a slightly denoised x_{t-1} from x_t. Equivalently to the forward process, this can be described by a Markov Chain:

$$p_\theta(x_{0:T}) := p(x_T) \prod_{t=1}^{T} p_\theta(x_{t-1}|x_t), \tag{2.29}$$

$$p_\theta(x_{t-1}|x_t) := \mathcal{N}(x_{t-1}; \mu_\theta(x_t, t), \Sigma_\theta(x_t, t)), \tag{2.30}$$

where $\mu_\theta(x_t, t)$ and $\Sigma_\theta(x_t, t)$ are the predicted Gaussian mean and the covariance matrix. When $\Sigma_\theta(x_t, t)$ is fixed to $\beta_t \mathbf{I}$ and $\mu_\theta(x_t, t)$ is described using the noise $\epsilon_\theta(x_t, t)$ predicted by the network, this can be expressed as:

$$p_\theta(x_{t-1}|x_t) := \mathcal{N}(x_{t-1}; \frac{1}{\sqrt{\alpha_t}}(x_t - \frac{1 - \alpha_t}{\sqrt{1 - \bar{\alpha}_t}} \epsilon_\theta(x_t, t)), \beta_t \mathbf{I}). \tag{2.31}$$

With these equations, Ho et al. [2020] were able to derive a simple loss function for training the diffusion model. They start from the variational lower bound of the negative log-likelihood:

$$\mathcal{L} = \mathbb{E}\left[D_{KL}\left(p(x_T|x_0) \| p(x_T)\right)\right] + \sum_{t \geq 1} D_{KL}\left(p(x_{t-1}|x_t, x_0) \| p_\theta(x_{t-1}|x_t)\right) - \log p_\theta(x_0|x_1), \tag{2.32}$$

with the Kullback-Leibler divergence D_{KL}. After several simplifications, they arrive at the following approximation for the loss function:

$$\mathcal{L}_{\text{simple}} := \mathbb{E}_{t, x_0, \epsilon}\left[\|\epsilon - \epsilon_\theta(x_t, t)\|^2\right], \tag{2.33}$$

which can be directly used for training the neural network as described in Algorithm 2.

Model Architectures. Denoising models are usually implemented using U-Net- [Ronneberger et al., 2015] or Transformer-based [Vaswani et al., 2017] architectures [Ho et al., 2020; Nichol and Dhariwal, 2021; Rombach et al., 2022; Peebles and Xie, 2023]. For their original diffusion model DDPM, Ho et al. [2020] used the U-Net-based PixelCNN [Salimans et al., 2017] and adjusted it by replacing some residual blocks [He et al., 2016] with self-attention blocks and by injecting diffusion time via Transformer

Algorithm 2 Training of diffusion model according to Ho et al. [2020]

for number of training iterations **do**
Sample initial image $x_0 \sim q(x_0)$
Sample timestep $t \sim \{1, \ldots, T\}$
Sample noise $\epsilon \sim \mathcal{N}(0, \mathbf{I})$
Update the diffusion network's weights (θ) by:

$$\nabla_\theta \left\| \epsilon - \epsilon_\theta \left(\sqrt{\bar{\alpha}_t} x_0 + \sqrt{1 - \bar{\alpha}_t} \epsilon, t \right) \right\|^2$$

end for

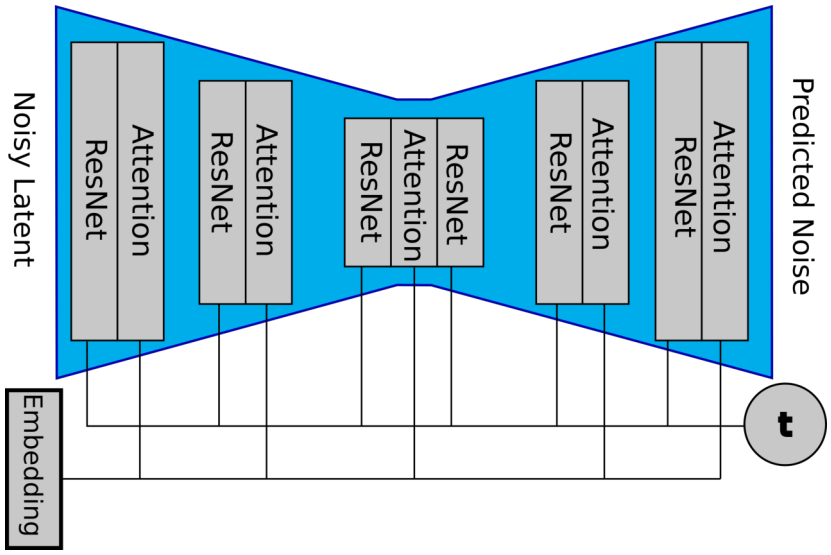


Figure 2.13: High-level overview of the typical U-Net architecture for diffusion models. During the reverse diffusion process, it receives the noise-modified input, which is passed through several ResNet and attention blocks to predict the noise for the next step. To influence image generation, conditioning can be passed as an embedding through cross-attention into the attention layers. Additionally, a timestep embedding (t) is inserted. Illustration adjusted from Po et al. [2023].

sinusoidal position embedding [Vaswani et al., 2017] into the residual blocks. Since then, various variations of this architecture have been proposed, such as increasing the number of attention blocks and their respective attention heads [Nichol and Dhariwal, 2021; Dhariwal and Nichol, 2021] or using BigGAN [Brock et al., 2019] residual blocks [Dhariwal and Nichol, 2021]. Moreover, Nichol et al. [2021] have leveraged the attention elements to inject text conditioning into the model. A high-level overview of a typical U-Net used in diffusion is presented in Figure 2.13.

Guidance and Conditioning. Similarly to the development in GANs and other generative approaches, one important direction of research has been to incorporate a condition c to steer the output. The earliest approaches to achieve class-conditional image generation with diffusion models passed c together with the time step encoding to the diffusion network [Nichol and Dhariwal, 2021; Zhou et al., 2021; Lyu et al., 2022]. Dhariwal and Nichol [2021] showed that using classifier guidance for conditioning can improve the image quality and allows for adjusting the strength of the influence of c. To achieve this, the scaled gradient of a pretrained classifier is added to the noise prediction of the diffusion model during sampling. Later, Ho and Salimans [2021] proposed classifier-free guidance, simplifying guidance by removing the dependence on a separate classifier. They employ a conditional denoising model $\epsilon_\theta(x_t, c)$ and replace the conditioning with a null label $\emptyset$ with a certain probability during training. In this way, it learns to also function as an unconditional model. During sampling, the output of the class-conditioned version of the model is added to that of the unconditional model, replacing the external classifier used for classifier guidance:

$$\hat{\epsilon}_\theta(x_t \mid c) = \epsilon_\theta(x_t \mid \emptyset) + s \cdot (\epsilon_\theta(x_t \mid c) - \epsilon_\theta(x_t \mid \emptyset)) \,, \tag{2.34}$$

where s is the guidance scale, which is often referred to as the CFG scale. Note that if the CFG scale is set to 1, this becomes the formula for unconditional diffusion.

Latent Diffusion. Stable Diffusion [Rombach et al., 2022], an open-source large-scale text-to-image model, builds on these advances. Its main improvement over its predecessors, such as DALL-E [Ramesh et al., 2021], is the usage of a pretrained autoencoder [Esser et al., 2020] to downsample the input to the latent space prior to passing it through the diffusion process. This reduction in dimensionality allows the model to focus its learning capacity on the semantics of the image data rather than barely perceptible high-frequency details contained in the pixel space. Therefore, training this latent diffusion model (LDM) is computationally more efficient.

Customization. Still, fine-tuning an LDM to customize it to specific needs requires large computational resources. An alternative approach is to employ an adapter, an additional network that connects to the original pretrained model [Mou et al., 2023; Zhang et al., 2023]. For example, IP-Adapter [Ye

et al., 2023] can be used to add image prompt capabilities to Stable Diffusion. It consists of a pretrained image encoder (CLIP [Radford et al., 2021]), which is applied to the prompt image, a small projection network to convert the embedding into the required dimensionality, and additional cross-attention layers to pass the image prompt into Stable Diffusion's U-Net [Ronneberger et al., 2015] via decoupled cross-attention. Only the projection network and the added cross-attention layers need to be trained, making this a compute-efficient way to adjust Stable Diffusion to specific needs.

Editing. Another important research direction is guided image synthesis. SDEdit [Meng et al., 2021] allows for guided image editing by first perturbing the image (or only a given region) with Gaussian noise and then using the standard reverse diffusion process. As the input image is not converted to random noise but only distorted to a certain degree, the output is guided by the coarse structures of the input image. The degree to which the original image is perturbed depends on the strength parameter t_0, which can range from zero to one.

Faster Sampling. One drawback of diffusion models is the slow generation. While other generative approaches, such as GANs, can create samples in a single step, diffusion models require a large number of network evaluations, leading to low efficiency. One way to speed up the generation process is by reducing the number of necessary sampling steps. The authors of DDIM [Song et al., 2021] achieve this by using a non-Markovian process, which can be used for reverse diffusion during sampling for models trained with the DDPM objective. Others distill existing models, e.g., using a teacher-student approach to create faster models [Salimans and Ho, 2022]. Song et al. [2023] improve upon this idea, demonstrating that their distilled Consistency Model allows high-quality image generation with a single step. Latent Consistency Models (LCM) [Luo et al., 2023a] transfers this idea to latent diffusion models, which can easily be applied to arbitrary pretrained models without training using LCM-LoRA [Luo et al., 2023b].

2.5.3 Implications of Memorization of Original Images

Even though GANs and diffusion models have been shown to generalize well and not merely reproduce the original data, there is a remaining risk of models

overfitting on a specific subset of the training images [Nagarajan et al., 2018; Li et al., 2023; Somepalli et al., 2022; Feldman and Zhang, 2020; Webster et al., 2019]. This *memorization* can severely impact the privacy of individuals appearing in the training data of generative methods [Otroshi-Shahreza and Marcel, 2024; Tinsley et al., 2020, 2022].

Its probability can depend on many factors, such as model size, training dataset size, or the amount of duplicates in the dataset. Feng et al. [2021] demonstrate that for StyleGAN2 and BigGAN the likelihood exponentially decreases with the number of samples in the training dataset. Somepalli et al. [2022] show that pixel-level reproduction for the diffusion model DDPM [Ho et al., 2020] is unlikely as long as the training set is sufficiently large. However, they also find the memorization probability specifically for the latent diffusion model [Rombach et al., 2022] trained on the LAION-5B dataset [Schuhmann et al., 2022] to be surprisingly high. This could be due to the large number of image duplicates or training specifics, such as the number of gradient updates being large enough to cause overfitting. Moreover, [Carlini et al., 2023] demonstrate that training data can be directly extracted from popular models like Stable Diffusion and Imagen [Saharia et al., 2022] and suggest that they are more prone to memorization than GANs.

Strategies to mitigate memorization include data deduplication or differential privacy (DP) [Dwork et al., 2006], clipping gradients and adding noise to them during training [Abadi et al., 2016]. This strategy has been successfully applied to GANs [Xu et al., 2019] and small-scale diffusion models [Chu et al., 2023], but can affect data utility.

Chapter 3

Related Work: Face De-Identification with Utility Retention

In this chapter, we discuss existing strategies for face de-identification in image data while retaining their usefulness for later applications. We first contextualize synthesis-based image anonymization, on which the two novel approaches introduced in this work are based, within the broader landscape of biometric privacy-enhancing techniques. Next, we introduce relevant datasets and evaluation strategies to quantify the performance of these approaches, followed by an in-depth discussion of the state of the art. Finally, we analyze possible applications for downstream tasks that rely on detailed facial features, as well as the current limitations of existing approaches.

3.1 Biometric Privacy-Enhancing Techniques for Images

There exist a multitude of techniques designed to mitigate the risks to personal privacy linked with sharing, storing, and processing biometric image data. Following the taxonomy introduced by Meden et al. [2021], these biometric privacy-enhancing techniques (B-PET) for images can be divided into three groups according to their point of application within a biometric recognition system: at the inference level, at the representation level, or at the image level (see Figure 3.1).
Inference-level techniques are utilized when data is employed for matching or classification purposes. These techniques ensure that the data is used solely for the intended purpose [Terhörst et al., 2020]. Representation-level

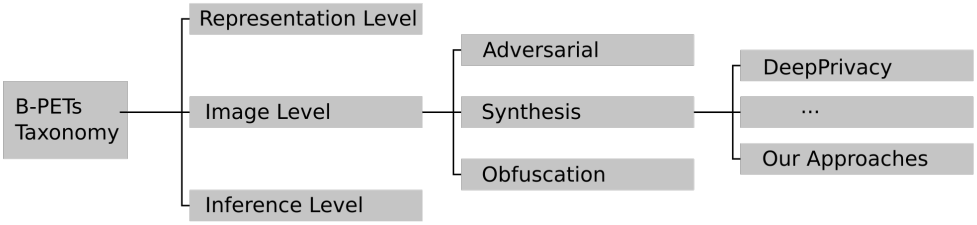


Figure 3.1: Taxonomy of biometric privacy-enhancement techniques following Meden et al. [2021]. Our approaches introduced in Chapter 4 belong to the group of synthesis-based techniques operating on the image level.

techniques are applied to a template representation obtained from the original image data and use strategies such as feature removal [Terhörst et al., 2019] or homomorphic encryption [Wingarz et al., 2022; Pulido-Gaytán et al., 2021]. Finally, image-level techniques often directly alter the image. They are well-suited for anonymizing images intended for human observers and do not impose constraints on the representation that can be extracted from it. This flexibility allows for a broad range of downstream applications, making them the focus of this study.

Image-level techniques can be further categorized into three distinctive classes (cf. Figure 3.1): adversarial, obfuscation and synthesis-based approaches. Adversarial approaches utilize adversarial perturbations, examples, or noise to modify the original image data in a manner that is often imperceptible to humans. Nevertheless, they can significantly reduce the performance of re-identification models [Chhabra et al., 2018; Gafni et al., 2019]. A concern with these approaches is that they typically cannot prevent identification by humans and only protect against specific models. Obfuscation techniques, on the other hand, comprising traditional methods like masking or heavily blurring privacy-sensitive regions, can prohibit recognition by humans as well as automatic identification. However, these methods substantially distort the visual data, reducing its utility for downstream analysis by humans or for computer vision applications that rely on facial details [Klomp et al., 2021; Hukkelas and Lindseth, 2023; Lee and You, 2024]. Addressing this limitation, synthesis approaches generate artificial image data that serves as a surrogate for the original face instead of obfuscating it. Thus, they offer enhanced

utility retention. Moreover, they are target-generic, differentiating them from adversarial approaches. To capitalize on these advantages, the two novel approaches we will introduce in Chapter 4 follow this concept.

3.2 Synthesis-Based Privacy Enhancement

In this section, we describe several recently developed state-of-the-art deep learning-based approaches of this group as well as the metrics and datasets used to evaluate them before introducing our two novel approaches in Chapter 4.

3.2.1 Datasets

We start by detailing the datasets we use in this work, all of which are publicly available.

LFW. Labeled Faces in the Wild (LFW) [Huang et al., 2008] contains 13,233 images of 5,749 celebrities sampled from the "Faces in the Wild" database [Berg et al., 2004], reducing labeling errors and image duplicates. It was

Figure 3.2: Examples of image pairs from the LFW dataset [Huang et al., 2008]. Matched pairs (top) contain two photos of the same person, while mismatched pairs (bottom) contain two images showing different people.

collected to support the development of face recognition in unconstrained environments with different facial expressions, lighting conditions, focus, resolution, occlusions and backgrounds. As it contains labels for pairs of images that show the same person (matched pairs) or two different persons (mismatched pairs), it is commonly used for evaluating face verification (see Figure 3.2). In recent years, it has been established as a benchmark for privacy protection of synthesis-based de-identification approaches [Maximov et al., 2020; Gafni et al., 2019].

CelebA. The CelebFaces Attributes (CelebA) dataset [Liu et al., 2015b] consists of 202,599 images with 10,177 unique identities. For all of our experiments, we use the aligned and cropped version of the test data split of CelebA, which contains 19,962 images. The dataset provides annotations on identity, landmark location, and binary attributes describing features such as head shape or expression. However, we only use it to illustrate the quality of faces anonymized with the approaches described in this work.

Biwi. The Biwi Kinect Head Pose Database [Fanelli et al., 2013] (hereafter referred to as "Biwi") is comprised of 14,934 images of 20 volunteers. These were recorded while the subjects were seated approximately one meter away from a Kinect, which was used to capture the ground truth head pose data. The recordings display head poses with Euler angles ranging between $\pm 75^\circ$ yaw and $\pm 60^\circ$ pitch.

JAFFE. This dataset contains 213 grayscale images with a size of 256×256 pixels of ten female Japanese volunteers acting out seven facial expressions: happy, sad, fear, disgust, anger, surprise and neutral [Lyons et al., 1998]. Images labeled as fear, disgust, or anger are not used in this study, as these emotions have been found to be challenging to classify on this particular dataset in previous research [Cho et al., 2020; Lyons et al., 1998]. This decision leaves us with 122 images, about 30 for each expression. Despite its modest size, the dataset is widely used as a baseline for emotion detection [Chen et al., 2020b; Cho et al., 2020].

AffectNet. AffectNet is a large-scale dataset comprising approximately one million facial images displaying various expressions [Mollahosseini et al., 2019]. These images were gathered from three search engines across six languages. Half of the dataset was manually labeled with the following emotion

categories: neutral, happy, sad, surprise, fear, disgust, anger, contempt, and a non-face class. As the test split is not publicly available, we use the validation split of this dataset to quantify retention of emotion after de-identification. We exclude images of the contempt and non-facial categories because the emotion detection model we use is not trained to classify these. This results in 3,497 images, about 500 for each emotion.

WIDER FACE. WIDER FACE [Yang et al., 2016] is a public dataset that was collected for training and testing face detection models. It was derived from WIDER [Xiong et al., 2015], which used event categories from Large Scale Concept Ontology for Multimedia (LSCOM) [Naphade et al., 2006] to query search engines such as Bing or Google for up to 3,000 images per category. For WIDER FACE, images without faces and near-duplicates were manually removed, resulting in a total of 32,203 images. On these, 393,703 faces were labeled with bounding boxes, tightly outlining the forehead, chin and cheek. For the purposes of this study, the training split, which contains 12,880 images and 159,393 faces, was used to train YOLOv8 (cf. Section 2.3.2) and DSFD models (see Section 2.3.4). The validation split with 3,226 images and 39,697 faces was used to evaluate the models, as the ground truth for the test split has not been publicly released. This approach is justified, as the validation data is never used to influence training and it is necessary to align with prior work for fair comparison [Klomp et al., 2021].

(a) (b)

Figure 3.3: Examples demonstrating the different levels of difficulty for the ground truth bounding boxes given by the WIDER FACE dataset [Yang et al., 2016]. Green: “Easy”, Blue: “Medium”, Red: “Hard”.

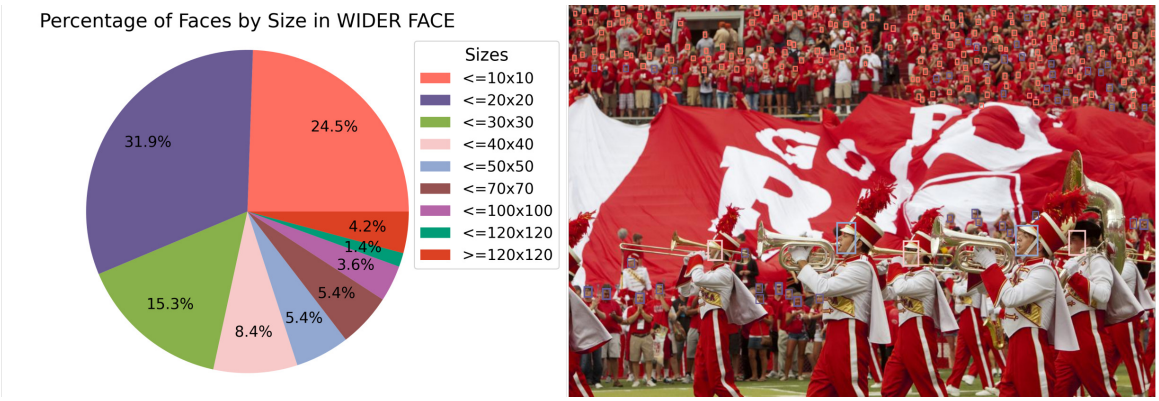


Figure 3.4: Left: the distribution of face sizes in the WIDER FACE dataset [Yang et al., 2016]. Right: image from WIDER FACE showcasing the different scales of faces appearing in this dataset. The boxes represent the given ground truth bounding boxes, with colors indicating sizes as given in the legend.

The faces of this dataset are typically divided into three levels of difficulty: "Easy," "Medium," and "Hard", which are based on their detection rate when using FaceBox [Zitnick and Dollár, 2014] (see Figure 3.3). Another important characteristic of the dataset is the distribution of the sizes of the ground truth bounding boxes. Figure 3.4 shows that the large majority of the faces are contained in boxes smaller than 30×30 pixels. The implications of this distribution for one of our novel approaches will be discussed in Section 4.3.

3.2.2 Evaluation Metrics

We use the datasets introduced in the preceding section to evaluate the performance of the synthesis-based approaches to de-identification discussed in this work. These are commonly evaluated regarding their ability to protect images against automated recognition (privacy protection), image quality and utility for downstream applications [Maximov et al., 2020; Hukkelås et al., 2019; Klomp et al., 2021; Hukkelas and Lindseth, 2023].

Privacy Protection. Following the work of Gafni et al. [2019] and Maximov et al. [2020], we quantify privacy protection with a modification of the procedure established for benchmarking face verification on Labeled Faces in the Wild [Huang et al., 2008]. To this end, we anonymize one of the images of each matched pair given by the dataset, but not the other. Mismatched pairs also stay unchanged. The two face recognition models, FaceNet and ArcFace (see Section 2.4), are employed to compute the distance between the

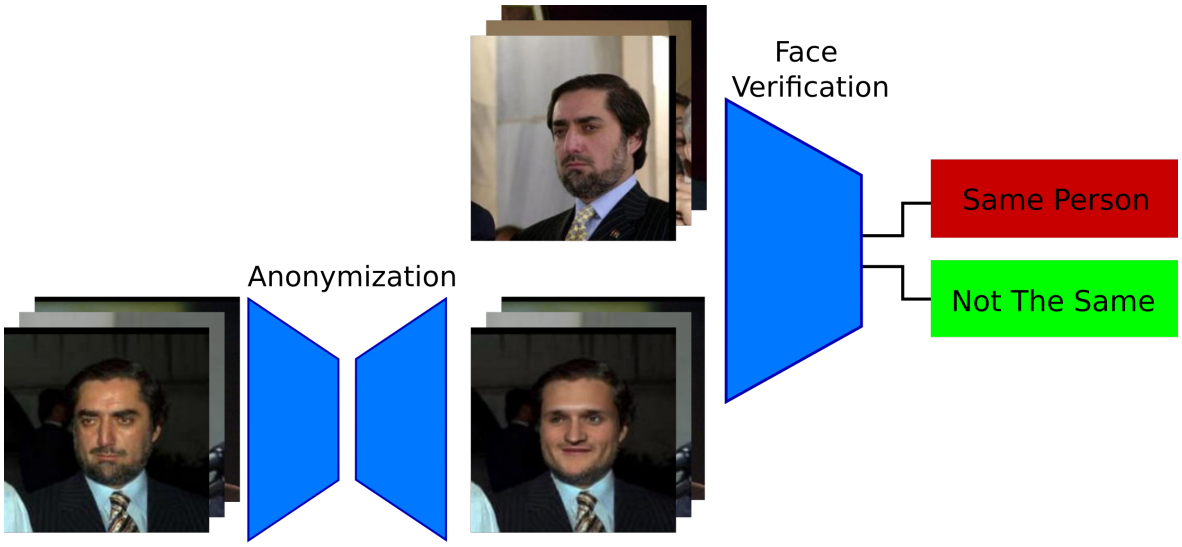


Figure 3.5: Evaluation procedure for privacy protection. First, one of the images of a matched pair from the LFW dataset (see Section 3.2.1) is anonymized using the approach we want to evaluate. Then, a face recognition model is used to perform face verification against the second image of the matched pair. If the anonymized image is not recognized as the same person, privacy protection is good.

two images of a pair in the embedding space. This distance is used in turn to decide whether they can still be recognized as the same person with the help of a threshold. This is visualized in Figure 3.5. From the ten subsets of the dataset, consisting of 300 matched and 300 mismatched pairs each, the aggregate performance of the face recognition model is evaluated. In each step of the cross validation scheme, nine of the subsets are used to determine the threshold distance in the face similarity space at which the False Acceptance Rate (FAR) is 10^{-3}. The True Acceptance Rate (TAR) is then measured on the last remaining subset. This is repeated ten times in total, each time changing the subset that is used for evaluation. Finally, the estimated mean TAR $\hat{\mu}_{TAR}$ is reported as given by

$$\hat{\mu}_{TAR} = \frac{\sum_{i=1}^{10} t_i}{10}, \tag{3.1}$$

where t_i is the TAR when using subset i for evaluation. Additionally, the standard error of the mean is calculated with

$$S_{err} = \frac{\hat{\sigma}}{\sqrt{10}}, \tag{3.2}$$

with the estimate of the standard deviation $\hat{\sigma}$, which can be determined using

$$\hat{\sigma} = \sqrt{\frac{\sum_{i=1}^{10}(t_i - \hat{\mu}_{TAR})^2}{9}}. \tag{3.3}$$

This metric is then employed to compare the privacy protection performance of different de-identification approaches. The lower the TAR, the better the protection against recognition.

Image Quality. Fréchet Inception Distance (FID) [Heusel et al., 2017] is often used to estimate the quality of generated images, as a good (low) FID value correlates with human perception of similarity. It builds on the idea of the Inception Score [Salimans et al., 2016] to use the image classification model Inception v3 (cf. Section 2.2) trained on ImageNet [Deng et al., 2009] to evaluate quality (see Figure 3.6). For FID, the image features calculated by the inception network after the last pooling layer are extracted. To them, two normal distributions are fit, one for the real and one for the generated

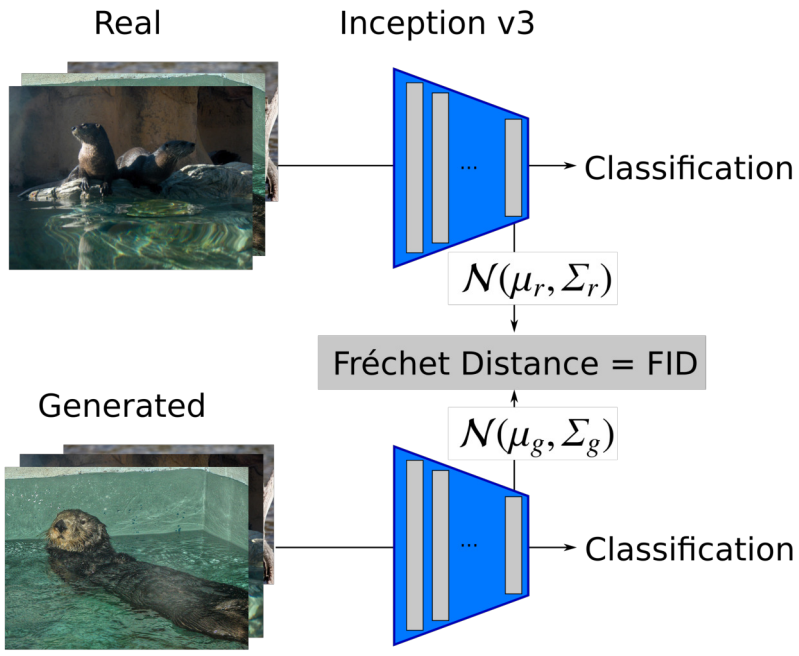


Figure 3.6: The Fréchet Inception Distance (FID) compares a real dataset with a synthetically generated one, estimating the quality of the generated images. First, both datasets are processed by an Inception v3 model and the feature vectors from the network's last pooling layer are extracted. Then, from all the feature vectors of each dataset, two multivariate Gaussian normal distributions are derived. The Fréchet distance between the distributions is called the FID, with lower distances indicating better image quality.

dataset. Then, the FID is calculated using the closed-form solution of the Fréchet distance for multivariate normal distributions [Dowson and Landau, 1982]:

$$FID = \|\mu_r - \mu_g\|^2 + \mathrm{Tr}(\Sigma_r + \Sigma_g - 2(\Sigma_r\Sigma_g)^{1/2}), \tag{3.4}$$

where μ_r and μ_g are the means of the distributions for the real and the generated dataset and Σ_r and Σ_g their respective covariances. Despite recent criticism of its dependence on ImageNet classes and the assumption that features can always be represented by a normal distribution [Betzalel et al., 2022; Jayasumana et al., 2023], it remains a standard metric for evaluating synthesis-based de-identification [Hukkelås et al., 2019; Maximov et al., 2020; Hukkelås and Lindseth, 2023].

An alternative metric that does not rely on features following a normal distribution is Kernel Inception Distance (KID) [Bińkowski et al., 2018], as it uses Maximum Mean Discrepancy (MMD) as a distance metric. We use it in addition to FID to obtain a more complete estimate of image quality.

Data Utility. While TAR, FID and KID are important to quantify the performance of de-identification methods, they do not quantify the utility of the resulting images.

Therefore, we use two additional metrics, the retention of head pose and of facial expression, to estimate the utility for downstream tasks connected to the analysis of humans. To measure head pose retention, we apply each de-identification method to the Biwi dataset and afterwards calculate the faces'

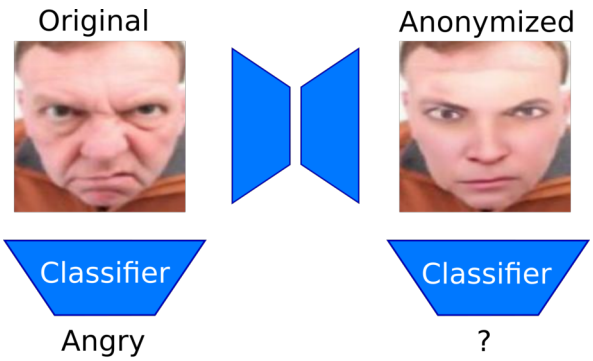


Figure 3.7: To evaluate utility retention, the images of a dataset labeled with expressions are anonymized with different de-identification approaches. Then, the accuracy of a machine learning-based expression classification model is calculated. The higher the accuracy after anonymization, the better the utility retention.

Euler angles using the approach of Ruiz et al. [2018]. We then compute the difference to the ground truth given by the dataset and compare this to the value calculated for the original data. The lower the Mean Absolute Error (MAE), the better.

The conservation of facial expression is judged by applying each de-identification approach to the JAFFE and AffectNet datasets, which contain labels for facial expressions. Then, we measure the accuracy of emotion detection models [Serengil and Ozpinar, 2021; Pham et al., 2021] before and after de-identification (see Figure 3.7).

Another way to evaluate data utility is to directly measure the implications for the downstream task when anonymized instead of original data is used. For example, Klomp et al. [2021] established measuring the utility for training a face detection model by comparing the performance of face detection models trained on original data to the same model trained on anonymized data (see Figure 3.8). To quantify the performance, the mean Average Precision (mAP) [Everingham et al., 2005] is computed using the precision and recall of a model considering only predictions above a given confidence threshold:

$$\text{Precision} = \frac{\text{True Positives (TP)}}{\text{True Positives (TP)} + \text{False Positives (FP)}} \tag{3.5}$$

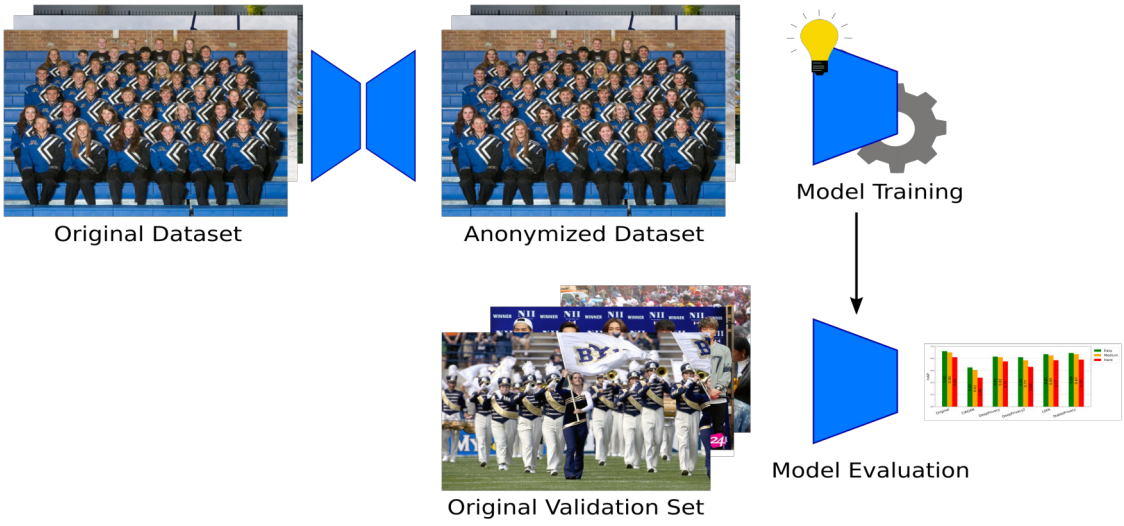
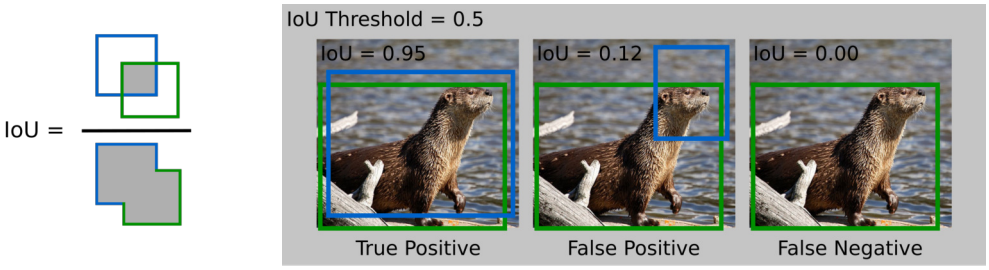


Figure 3.8: Evaluation of the data utility of anonymized data for training face detection models by directly measuring the performance change in comparison to using real data.



(a) (b)

Figure 3.9: (a) Visualization of IoU. The green bounding box represents the ground truth; the blue bounding box is a prediction made by an object detection model. The area of overlap between ground truth and prediction determines the numerator of the IoU, the combined area (area of union) is the denominator. (b) If the IoU between prediction and ground truth is higher than the threshold, it is considered a true positive (TP); if it is lower, it is a false positive (FP). In case no prediction has been made that overlaps with the ground truth, this is a false negative (FN).

$$\text{Recall} = \frac{\text{True Positives (TP)}}{\text{True Positives (TP)} + \text{False Negatives (FN)}}. \tag{3.6}$$

The definition of a true positive in object detection depends on the Intersection over Union (IoU)

$$\text{IoU} = \frac{\text{Area of Overlap}}{\text{Area of Union}}, \tag{3.7}$$

of the bounding box predicted by the model with the ground truth bounding box (see Figure 3.9). For our evaluations, we set the required IoU to 0.5. The calculation of precision and recall is repeated, decreasing the confidence threshold from one to zero in steps of 0.001 to create the precision-recall curve. The average precision (AP) is then calculated by integration:

$$AP = \int_0^1 P(r)\, dr. \tag{3.8}$$

To get the mean Average Precision this is repeated for all object classes we want to detect and the average is computed:

$$mAP = \frac{1}{N} \sum_{i=1}^{N} AP_i. \tag{3.9}$$

However, as there is only one class in face detection, the mAP is equal to the AP. The value ranges from zero to one, with one indicating the perfect score.

3.2.3 Current Synthesis-Based De-Identification Approaches

We employ the metrics introduced in the previous section to assess synthesis-based de-identification approaches. Existing approaches are based on a broad spectrum of technical solutions. Some rely on standard image processing techniques to automatically replace faces or individual facial components with those of real donors [Xu et al., 2015; Bitouk et al., 2008; Mosaddegh et al., 2014]. Others employ statistical models [Gross et al., 2005, 2006; Newton et al., 2005; Du et al., 2014; Meng et al., 2017], such as the k-same method that identifies the k most similar faces in a video and substitutes each of them with their common averaged face [Newton et al., 2005]. However, in this work we focus on deep learning-based approaches, as these models typically create more natural-looking images by leveraging the recent developments in the field of generative machine learning, such as Generative Adversarial Networks (see Section 2.5.1) or diffusion models (cf. Section 2.5.2), which can synthesize photo-realistic faces [Karras et al., 2020, 2021a; Rombach et al., 2022].

The de-identification techniques to which we compare our novel approaches, *Conditional Identity Anonymization GAN* (*CIAGAN*) [Maximov et al., 2020], *DeepPrivacy* [Hukkelås et al., 2019], *DeepPrivacy2* [Hukkelås and Lindseth, 2023] and *Latent Diffusion Face Anonymization* (*LDFA*) [Klemp et al., 2023], naturally blend these generated faces with the background. To achieve this, the characteristics of the original image have to be considered.

Keypoints-Based Approaches: CIAGAN and DeepPrivacy. CIAGAN and DeepPrivacy incorporate a masked version of the original image together with its facial landmarks or keypoints to guide the synthesis process by training a GAN with a U-Net-based [Ronneberger et al., 2015] generator. While these two approaches are very similar on a high level (see Figure 3.10), they differ in several details.

DeepPrivacy adopts a GAN architecture inspired by proGAN (cf. Section 2.5.1), replacing the original unconditional generator with a U-Net-based design. This modification enables the model to be conditioned with keypoints and the masked-out image. Training follows the progressive growing technique of Karras et al. [2018], with both the generator and the discriminator initialized with an 8×8 pixel resolution, which is doubled multiple times

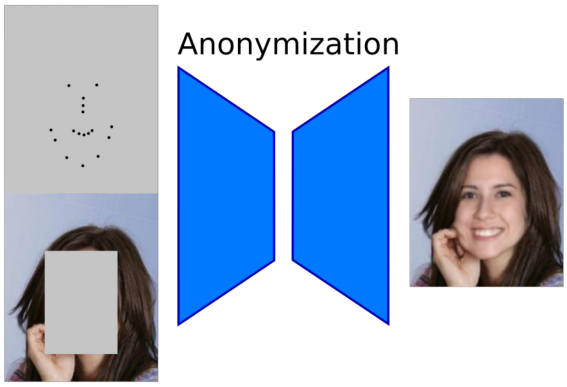


Figure 3.10: On a high level, the anonymization process of CIAGAN and DeepPrivacy is similar. Both use a masked version of the original image and facial landmarks extracted from it as input for a U-Net-shaped generator of a GAN to generate anonymized images. In detail, the approaches differ significantly in the number of facial landmarks or keypoints needed, the approach to generate the masked area, as well as the GAN training and architecture.

by appending 3×3 convolutional layers to the network until the target output size is reached. The model is trained on the Flickr Diverse Faces (FDF) dataset [Hukkelås et al., 2019] containing 1.47 million human faces with at least 128×128 pixels for 40 million iterations. As an input to the generator, DeepPrivacy uses seven keypoints at the ears, the center of the eyes, the nose, and the shoulders provided by Mask R-CNN [He et al., 2017]. While these can be reliably extracted for most images, the dependence on shoulder keypoints leads to low-quality outputs for tightly cropped faces. During training, keypoints are also provided to the discriminator, improving pose retention. For the generator's second input, a masked version of the original image, DeepPrivacy leverages Dual Shot Face Detector (see Section 2.3.4) to locate the face region and replaces it with random noise.

The U-Net generator employed by CIAGAN is based on the design of Shelhamer et al. [2014] and can be conditioned similarly. However, instead of seven keypoints, it employs 27 of the 68 landmarks provided by dlib [King, 2009], outlining the jaw, nose, and mouth. This choice significantly influences utility retention and privacy protection. As the landmarks contain information on head positioning and expression, they help to preserve these attributes and to generate faces that blend more naturally into the original background. On the other hand, they carry information on the identity, which can leak to the anonymized image. Moreover, relying on detailed landmarks also

affects the type of images the approach can be applied to, as they often cannot be extracted for low-resolution or otherwise challenging faces. For the second input, CIAGAN, unlike DeepPrivacy, only masks the area within the detected landmarks instead of a bounding box for the face and inpaints it with monochromatic pixels. Another important difference is that CIAGAN relies on an additional one-hot identity guidance vector, encoding the output identity, as an input to the generator. This helps steer the network away from the landmarks, avoiding to reproduce the original face too closely. In addition, it can be used to create similar-looking synthetic faces for creating temporally consistent videos when the same identity guidance vector is given to the generator for each frame. Conversely, DeepPrivacy does not control the output identity, therefore creating different faces in each frame when applied to videos.

Keypoints-Free Approaches: DeepPrivacy2 and LDFA. In contrast to DeepPrivacy and CIAGAN, DeepPrivacy2 and LDFA are keypoints-free approaches. Thus, they can anonymize faces even when no keypoints or landmarks can be detected.

DeepPrivacy2 was originally designed for full-body anonymization, only resorting to face anonymization if the required Continuous Surface Embeddings (CSE) of the entire body cannot be obtained. In this thesis we only consider its face de-identification functionality to allow for a fair comparison to the other approaches. It improves upon DeepPrivacy in terms of image quality by replacing the previous GAN with SG-GAN [Hukkelås et al., 2022], a StyleGAN-based architecture (compare Section 2.5.1) that they adjusted for faces and trained on the FDF256 dataset [Hukkelås and Lindseth, 2023]. Similarly to DeepPrivacy, a masked-out version of the original image is used as an input to guide the anonymization. For this, the standard StyleGAN generator is exchanged with a U-Net generator, allowing image-to-image translation.

LDFA was developed for the anonymization of vulnerable road users (e.g., cyclists or pedestrians) for tasks related to intelligent transportation systems, such as training a transformer-based semantic segmentation model or performing face detection on anonymized data without prior training on de-identified images. Unlike all other approaches, it does not rely on GANs and instead employs Stable Diffusion (cf. Section 2.5.2) to generate faces (see Figure 3.11).

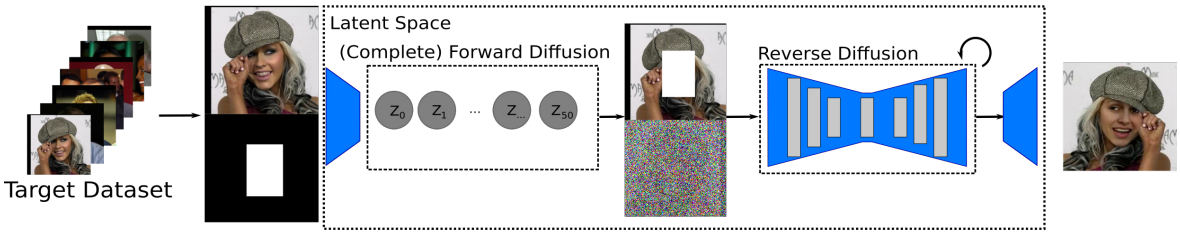


Figure 3.11: Overview LDFA. Faces detected by RetinaFace are the targets for anonymization. Inpainting is applied to the area of the detected bounding box (white), the additional 32 pixel around the face provide context to improve image quality. After 50 steps of forward diffusion, adding Gaussian noise to the image, the result is used as a starting point for the reverse diffusion process together with the masked-out context.

The simple, yet effective two-step pipeline starts by detecting face bounding boxes using RetinaFace [Deng et al., 2020]. The area of the image where the face has been detected, extended by 32 pixels to each side, is passed into Stable Diffusion. Before the reverse diffusion process starts to inpaint the masked-out area of the original, the image is subject to 50 forward diffusion steps, each adding Gaussian noise. This heavy distortion is meant to avoid a close reconstruction of the original. Yet it also removes structural cues, for example, regarding occlusions within the bounding box, making it harder to produce realistic transitions between the original background and the inpainted area. Moreover, despite this measure, LDFA cannot provide more privacy protection than DeepPrivacy or CIAGAN, even though they rely on explicit guidance from keypoints or landmarks.

3.2.4 Applications and Limitations of Previous Work

The approaches described in the previous section can significantly facilitate the sharing, storage and processing of datasets for academic and commercial use by ensuring regulatory compliance without compromising data utility. However, currently no existing solution provides perfect utility retention and is applicable to every possible downstream use case. In the rest of this section, we first present possible application scenarios and then discuss how far the approaches introduced previously can satisfy their specific requirements.

Sharing, Analyzing and Storing Videos for Complete Semantic Understanding. One scenario is to apply anonymization to images or videos from sensitive domains such as surveillance or patient data that still require detailed semantic understanding of the captured scenes. For example, Wilson et al. [2022] discuss applications for mitigating privacy risks created by sharing videos of patients for telehealth, remote training for clinicians or dataset sharing for computational research, specifically for the assessment of autism symptoms in children. For this application, anonymized data would have to retain details such as facial expressions as well as gaze and be applicable to video. Another use case is to de-identify surveillance footage to reduce the risks of an unauthorized attacker viewing the material and to increase acceptance from the monitored individuals. For instance, Ravi et al. [2021] discuss employing such techniques for residents of an Active and Assisted Living facility who have given consent to being monitored by the home's personnel for safety reasons. Here too, applicability to video and retention of expressions could be required to correctly assess potentially dangerous situations.

Training Machine Learning Models for Downstream Tasks. Another scenario is to anonymize datasets meant for training machine learning models prior to making them publicly available to the scientific community or companies. Many computer vision tasks, such as face detection, keypoint detection, action recognition or people tracking, do not depend on privacy-sensitive features identifying individuals. Instead, they rely on characteristics such as generic human shape and texture (face detection, people tracking) or head position and expression (keypoint detection, action recognition). Therefore, these tasks can be performed on and learned from the data downstream, even after appropriate anonymization. Among these applications, face detection probably has the least stringent prerequisites on anonymization to preserve details such as keypoints or expression, but requires robustness to crowded scenes, various types of occlusions or complicated lighting conditions. Moreover, the diversity of the original dataset has to be retained.

Applicability of Previous Work. Of the approaches discussed above, only CIAGAN seems applicable to the first scenario, as it works on videos and aims to retain detailed facial structures by utilizing a substantial number of

landmarks. However, CIAGAN often creates low-quality images for challenging head positions and occlusions and cannot retain complex expressions or gaze. Consequently, in the next chapter, we will introduce a novel, more robust approach using denser landmarks to conserve even more detail without significantly reducing privacy protection.
The second use case can, in principle, be addressed by all of the approaches. However, while Klomp et al. [2021] show that anonymized training data can be employed to train high-quality face detection models, the low quality of the synthesized faces when landmarks or keypoints cannot be extracted for CIAGAN or DeepPrivacy and the comparatively weak privacy protection of LDFA and DeepPrivacy2 limit their usefulness. Thus, we will present a second keypoints-free approach, improving upon both privacy protection and data utility retention compared to existing approaches.

Chapter 4

Two Novel Approaches for Synthesis-Based Privacy Enhancement

In this chapter, we present two novel approaches for synthesis-based privacy enhancement that were developed in the context of this thesis [Leibl et al., 2023; Leibl and Mayer, 2024]. They are based on a similar idea, namely using synthetically generated source faces to guide the anonymization process, but are optimized towards different objectives, as discussed in the preceding chapter. The first approach, which we call *DetailedPrivacy* in this thesis, aims to retain fine-grained facial features such as head position, gaze and detailed expressions.

The second approach, *StablePrivacy*, also uses a source library to guide de-identification, but refrains from using facial keypoints to retain detailed features. Instead, it employs rough structural guidance. Therefore, it is applicable to images where no or not all keypoints can be detected due to small face size, occlusions or other challenging conditions. While this approach cannot transfer details such as the exact expression to the anonymized image, it can preserve more coarse features such as a realistic human appearance. Images anonymized with this approach can, for example, be used as training data for face detection models.

We start with providing an overview of the approaches and comparing them to each other in Section 4.1. In Sections 4.2 and 4.3 we evaluate our novel approaches against state-of-the-art approaches for their respective use case.

4.1 Overview of the Approaches

4.1.1 DetailedPrivacy: De-Identification Retaining Facial Details

DetailedPrivacy is our approach for anonymizing faces focusing on retaining detailed facial features. It combines FSGANv2 (see Section 2.5.1), a powerful face swapping method, with a carefully created library of synthetic source faces and an automatic source selection mechanism designed to balance two competing objectives: safeguarding privacy and maintaining data utility.
The first component of our approach is the source library. The idea is to create a collection of images from which we can later choose the most suitable source, because, contrary to face swapping, we are not restricted to using the image of a specific real person. Moreover, employing only synthetic source faces ensures that the generated images cannot infringe upon the privacy rights of a third person. To build this library, we generated 100 faces with StyleGAN2 (cf. Section 2.5.1) and optimized them for their use in DetailedPrivacy by creating additional views of the synthetic faces from different angles. This is implemented by applying dlib's [King, 2009] facial landmark detector to extract 68 facial landmarks and computing the rotation matrix of a given face. From this matrix, an attribute vector is generated using the method developed by Meißner et al. [2022], which can be employed to iteratively adjust the yaw angle for each of the 100 seeds to the required head position. This way, for each source, we create five images at different angles (approximately 25°, 15°, 0°, −15°, −25°), which we finally resized to 256×256 pixels. Examples from the resulting source library are given in Figure 4.1. The second component of our approach is the automated selection of a source image for each target, as handpicking is unfeasible when anonymizing large-scale datasets. Our approach relies on two parameters influencing this choice: face similarity measured by Euclidean distance in FaceNet's embedding space (see Section 2.3) and head pose, estimated using Hopenet [Ruiz et al., 2018]. While a source image that closely resembles the target may better preserve data utility, opting for a face with greater dissimilarity may yield better privacy protection. The source pose can influence anonymization as large differences to the target lead to more significant adjustments by FSGANv2, which can cause identity leakage and

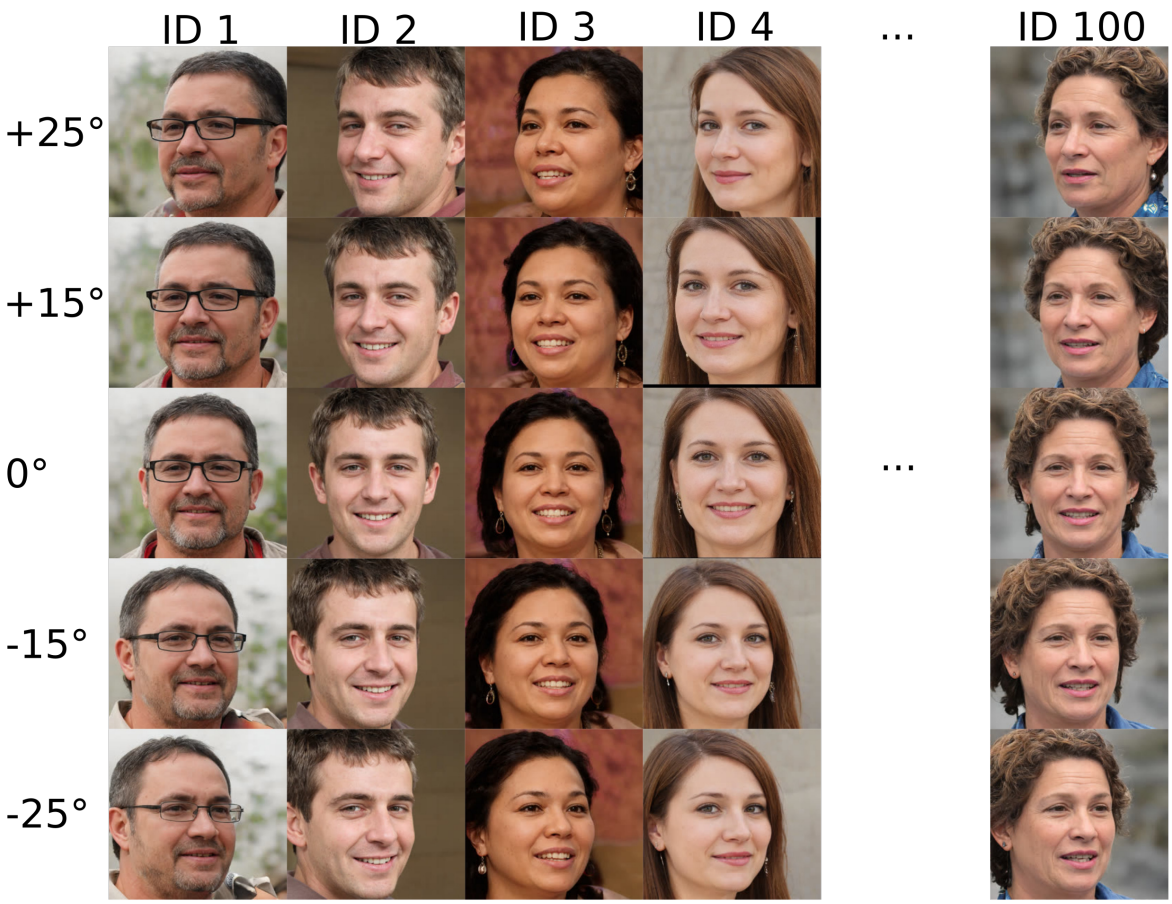


Figure 4.1: DetailedPrivacy's source library is a collection of 100 synthetic face images generated with StyleGAN2 (see Section 2.5.1). For each face, five different views were created depicting the same face at different Euler angles (yaw angle: 25°, 15°, 0°, −15°, −25°).

reduced texture quality, as discussed by Nirkin et al. [2022]. Empirically, we found that choosing a face with a pose roughly similar to the target while also maximizing facial dissimilarity provides the best balance between privacy protection and data utility retention. To implement such a selection process, we first identify the three candidate faces from our source library with the largest FaceNet embedding distance from the target. We randomly select one of them to guide the de-identification. This avoids a unique mapping that could be reversed by an attacker trying to undo the anonymization. For the chosen candidate, we use the generated view that depicts the face at the pose most similar to the target's. The benefits of this selection process are further quantified and discussed in Section 4.2.2. Finally, we pass the source as an input to FSGANv2, which leverages 98 landmarks from the target

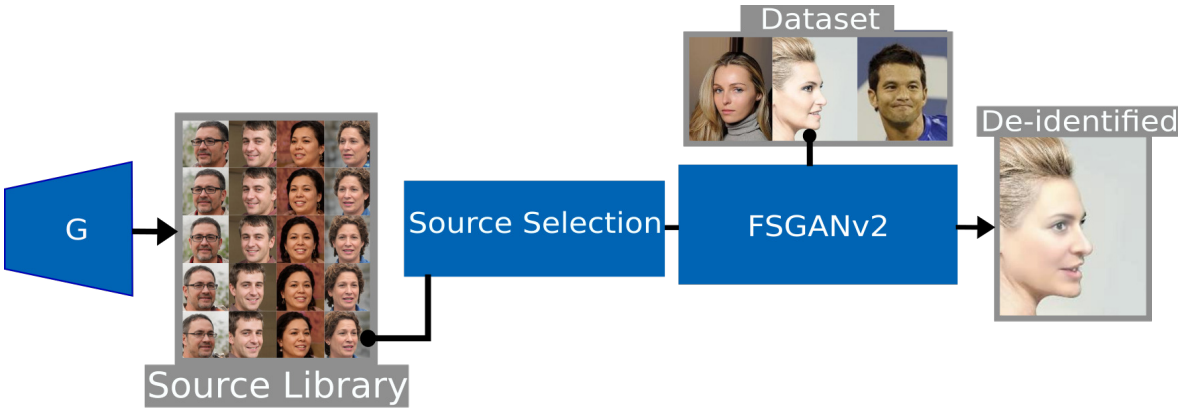


Figure 4.2: Overview of DetailedPrivacy. First, a suitable source library of synthetic faces that do not infringe privacy rights is created using StyleGAN2. The source selection process chooses a suitable image from the source library for each target image, which we want to anonymize. The source face is replaced by the target using the face swapping approach FSGANv2 (cf. Section 2.5.1). The resulting face image is de-identified, exhibiting little similarity to the original face while maintaining key attributes like pose and expression.

face determined by HRNet [Wang et al., 2021a] to retain the head position and facial expression for the anonymized image. A visual overview of our approach is given in Figure 4.2.

De-identification of Videos. With some minor modifications, our approach extends naturally to videos. For this, maintaining a consistent real-to-fake identity mapping [Maximov et al., 2020] is crucial for ensuring coherence across frames. Changing the fake identity for a given real person during a video sequence would interfere with tasks performed on the de-identified video. To enforce consistency, we use our source selection process on the first frame to assign a synthetic source face to each target face based on its distance in the similarity embedding space. For the rest of the video, we choose only from the five views generated for that source when optimizing for pose similarity.

4.1.2 StablePrivacy: Robust De-Identification with Strong Privacy Protection

StablePrivacy, our second approach for synthesis-based de-identification, relies on Stable Diffusion (`sd-v1-5-inpainting` [1]) for generating realistic inpaintings of the facial region that we want to anonymize. It does not require keypoints to guide face generation. Instead, a noise-modified version of the original image is employed, utilizing the SDEdit technique, along with a source face passed to Stable Diffusion using the IP-Adapter with pretrained weights (`ip-adapter_sd15` [2]). Even though previous work has explored keypoints-free de-identification [Hukkelås and Lindseth, 2023; Klemp et al., 2023], our approach provides uniquely strong privacy protection due to the usage of source faces as well as robustness in complicated scenes because of structural guidance from the target image. A graphical overview of our approach is shown in Figure 4.3.

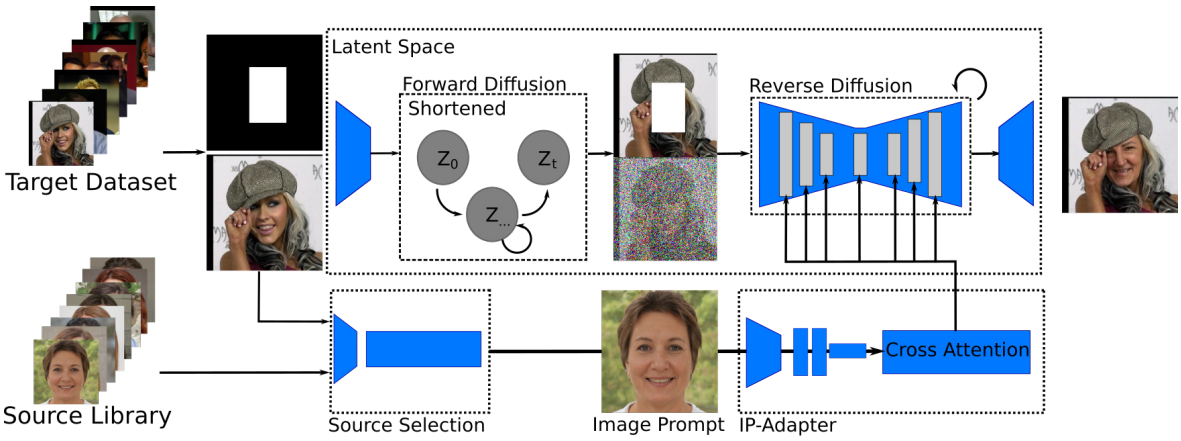


Figure 4.3: Overview of StablePrivacy. It starts by masking the facial region of the original image and encoding both the masked and the original image into Stable Diffusion's latent space. Then, Gaussian noise is applied to the original image, maintaining some of the structural details with a shortened forward diffusion process (strength = 0.7) using SDEdit (cf. Section 2.5.2). The masked as well as the noisy image are then passed to Stable Diffusion's U-Net for reverse diffusion. Simultaneously, a synthetic source face, selected for minimal similarity to the target, is processed by the IP-Adapter as an image prompt. Guided by these inputs, Stable Diffusion inpaints the previously masked region, creating an anonymized output image.

[1] https://huggingface.co/runwayml/stable-diffusion-inpainting/blob/main/sd-v1-5-inpainting.ckpt
[2] https://huggingface.co/h94/IP-Adapter/tree/main/models

Similarly to DetailedPrivacy, we create a library of synthetic source images with StyleGAN2 to ensure that no real individual's privacy is compromised. However, the dataset is significantly larger, containing 1,000 faces. Moreover, the images were manually curated, discarding faces with excessive occlusions by hair, glasses or other objects and balancing the gender distribution by choosing equal numbers of female and male faces. Additionally, instead of relying on multiple views, we adjusted the Euler angles of the heads (yaw angle) to approximately 0 degrees [Meißner et al., 2022] to ensure the image prompts contain sufficient relevant features to guide the identity transfer.
Inspired by the source selection process of DetailedPrivacy, StablePrivacy automatically chooses a source from our library that is sufficiently different from the target. This is achieved by requiring a minimum distance in the face similarity space between the source and the target, which is calculated using FaceNet (compare Section 2.4). The threshold distance is empirically determined and fixed to 1.6 for all experiments.
Once a suitable source is chosen, it is processed through IP-Adapter's projection network and passed into the attention layers of Stable Diffusion's U-Net using decoupled cross-attention. Its influence on the output image is regulated by the classifier-free guidance (CFG) scale.
To define the modification area, our approach requires bounding box coordinates outlining the face. These coordinates can be obtained either from a face detection model or, when available, from ground truth annotations provided by the dataset. Optionally, the size of the inpainting area can be increased by an additional margin around the bounding box to ensure even weakly identifying features, such as the hair, are changed.
When an image contains multiple faces that need to be anonymized, our approach handles them sequentially. For each face, the associated bounding box is extended by 100 pixels to each side (see Figure 4.4, large blue boxes) and this area is cut out and passed to StablePrivacy. As the region around the face provides context for the inpainting, it improves image quality. After anonymization, only the area modified by inpainting (see Figure 4.4, small red boxes) is used to replace the corresponding area in the original image.
In order to guide the inpainting process for better utility retention, two versions of the original image are prepared as input, limiting the identifying

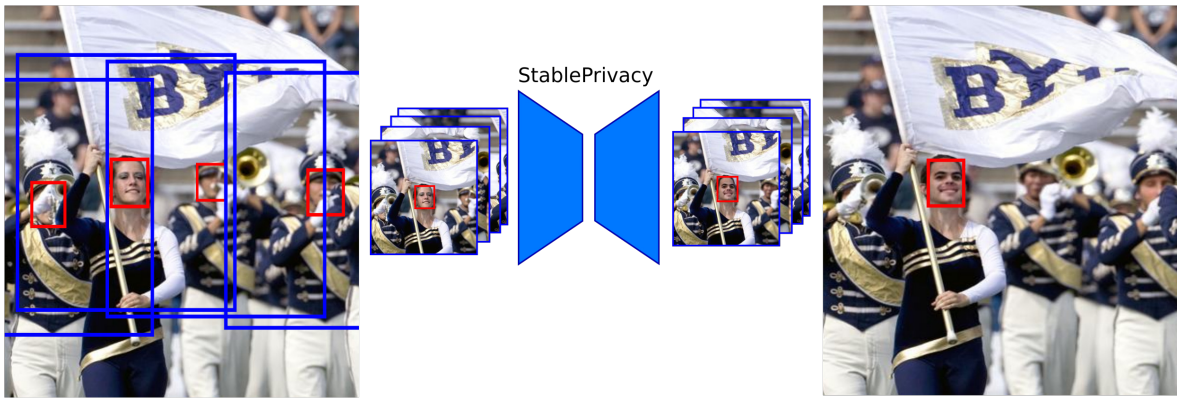


Figure 4.4: Image from WIDER FACE demonstrating anonymization of multiple faces in a single image. Individual image regions (larger blue boxes), each outlining one face (smaller red boxes) for anonymization, are cut out and sequentially passed to StablePrivacy. After anonymization, only the inpainted area (red box) is used to replace the corresponding area in the larger original image.

features they contain. In the first, the face is blacked out. In the second, it is superimposed with Gaussian noise, which can be regulated through a strength parameter (see Section 2.5.2) that ranges from zero (no noise) to one (only noise). The more noise is added, the less identity information can leak to the output image, yet ensuring the underlying structure remains vaguely intact can benefit robustness and utility retention. After empirical validation, which we will discuss in detail in Section 4.3.4, we found a value of 0.7 works best for faces of a reasonable size. Only if faces are smaller than 30×30 pixels, we find it acceptable to reduce the strength to 0.5, improving structural guidance at the expense of privacy protection. However, as the small size of these faces already limits recognition, less stringent protection is required. This choice is validated through several experiments in Section 4.3.4.
Apart from the strength, other parameters like the size of the inpainting area and the CFG scale affect the trade-off between privacy and data utility of our approach. Their choice is also discussed in Section 4.3.4.

4.1.3 Comparison of the Novel Approaches

After outlining our two novel approaches, DetailedPrivacy and StablePrivacy, in the previous sections, we compare them directly, illustrating their similarities and highlighting the differences that help them to excel at their specific

use case.

Both approaches employ similar strategies for de-identification, each using synthetic source faces in combination with structural cues from the original image to guide the anonymization. In DetailedPrivacy, the structural information consists of landmarks extracted from the target face. These contain detailed information about head position as well as expression and can be used to create temporally consistent videos. However, landmarks can leak identity to the output and cannot be extracted for challenging images or low-resolution faces. Therefore, their usage poses limits to privacy protection and robustness. In contrast, StablePrivacy relies on structural cues from the noise-modified original image, which only holds rough information on head position and expression, but additionally includes information about the structure of facial occlusions and can be obtained for all images. While this approach facilitates realistic anonymization under difficult conditions, it also tends to leak identifying features, as discussed in Section 4.3.4.

To counteract identity leakage, both approaches rely on a source face from an image library to steer the inpainting model away from creating faces too closely resembling the original. However, the structure of the source libraries differs, as they are chosen to work well with their specific inpainting approach. For DetailedPrivacy, five different views of each source are generated, as this reduces the number of iterations necessary to adjust the pose of the source face to that of the target with FSGANv2. In contrast, StablePrivacy does not require multiple views. Instead, faces are frontalized to avoid one-sided depictions that offer limited information for other angles.

Moreover, the source selection process in both approaches is very similar, using FaceNet to find source faces that are sufficiently different from the original. Our ablation studies in Sections 4.2.2 and 4.3.9 show the importance of this step for privacy protection. However, unlike StablePrivacy, DetailedPrivacy additionally requires selecting the view of the source that aligns best with the target position.

Finally, the two approaches differ in the inpainting method they use. While DetailedPrivacy relies on a more lightweight GAN-based approach, StablePrivacy uses a latent diffusion model, which is much larger. Therefore, DetailedPrivacy can be used with a smaller GPU, making it more accessible for many

users.

4.1.4 Differentiation from Face Swapping and Training on Synthetic Data

Before we continue with the evaluation of our novel approaches against the state of the art, we first clarify the distinction between de-identification, face swapping and the generation of fully synthetic data.
Although technically similar, our synthesis-based de-identification approaches differ from face swapping as they are optimized for a distinct objective. Face swapping is designed to convincingly replace the target's face with that of a specific real person, effectively placing the source individual into a scene they were not in. However, these methods typically lack safeguards against identity leakage of the target face to the final image, increasing the risk of re-identification. In comparison, our approaches focus less on the source and more on protecting the privacy of the target.
Additionally, face swapping is generally designed for modifying a limited number of images or videos. In such cases, users can manually select source and target faces that already share similarities or even fine-tune models for specific individuals. This flexibility allows for the creation of artificial images that are often indistinguishable from real ones.
In contrast, de-identification is intended for large-scale datasets and prioritizes anonymization over realism. Therefore, de-identification requires an entirely automated selection process that ensures privacy protection. The main objective is not to create visually undetectable swaps but to maximize privacy by preventing identity leakage while preserving data utility.
Another important difference exists between de-identification and fully synthesizing data [Joshi et al., 2022]. Both can be used as a substitute for real data in training machine-learning models for the analysis of humans. Indeed, several studies have explored generating training data entirely using generative machine-learning models, such as GANs, for applications like face recognition [Boutros et al., 2023, 2022; Kolf et al., 2023] and facial expression classification [Niinuma et al., 2020]. However, this approach is not suitable for tasks that require processing complex scenes. Crowded environments, global lighting conditions and occlusions, etc., are difficult to synthesize, yet

essential for high-quality training datasets in tasks like face detection.
Due to these challenges, we focus on de-identification, only changing the relevant facial regions while preserving the original scene context rather than generating entirely synthetic images.

4.2 Evaluation of DetailedPrivacy for Retaining Facial Details

4.2.1 Experiments

We start with the evaluation of DetailedPrivacy. Examples of faces anonymized with this approach are shown in Figure 4.5. In this section, we only consider keypoints- or landmarks-based de-identification tools, i.e., CIAGAN and DeepPrivacy, for comparison. We do not include the keypoints-free techniques LDFA, DeepPrivacy2 or StablePrivacy, as they cannot retain detailed features.

Figure 4.5: Examples from the CelebA dataset (see Section 3.2.1) anonymized with DetailedPrivacy. It can handle difficult lighting, partial occlusions and a variety of head positions, conditions other approaches often struggle with. While some of the images have a somewhat artificial look, convincing realism to a human observer is not the main focus of de-identification. Instead, we focus on privacy protection and utility retention.

De-ID Method	FaceNet (↓) [%]	ArcFace (↓) [%]	KID (↓)	FID (↓)
Original	98.60 ± 0.76	96.13 ± 1.81	N/A	N/A
Face Pixelization 16 × 16	0.56 ± 1.67	0.33 ± 0.26	0.0417 ± 0.0012	43.09
CIAGAN	3.40 ± 0.65	5.83 ± 1.97	0.0105 ± 0.0007	13.30
DeepPrivacy	10.90 ± 1.93	6.63 ± 2.12	0.0014 ± 0.0002	2.37
DetailedPrivacy	9.03 ± 1.01	11.47 ± 2.25	0.0146 ± 0.0008	13.26

Table 4.1: Performance for privacy protection (TAR measured using FaceNet or ArcFace) and image quality (KID, FID) evaluated on LFW for keypoints-based synthesis approaches for de-identification compared to pixelization serving as a baseline. Lower TAR implies better privacy protection. Lower values for KID and FID indicate better image quality.

Instead, these will be analyzed in Section 4.3 in the context of a different use case.

Basic Image Quality. The basic image quality of de-identified images is quantified with KID and FID as discussed in Section 3.2.2. Table 4.1 shows the quality of our pipeline compared to the original unaltered LFW dataset, face pixelization as a baseline traditional anonymization technique and to the synthesis-based de-identification approaches CIAGAN and DeepPrivacy. While we achieve significantly better KID and FID than the baseline method, CIAGAN and DeepPrivacy perform even better. However, these metrics only capture image quality across the entire dataset and do not account for specific scenarios. For instance, our approach often produces more realistic outputs for faces viewed at large angles or in challenging poses. As exemplified in Figure 4.6, our approach introduces fewer artifacts on a profile-oriented face (row 4) and handles the difficult occlusion by the shoulder (row 5). Moreover, even though achieving a certain level of image quality is beneficial, the main objective of de-identification approaches is more retaining details of the face, like the exact pose or facial expression, which cannot be quantified by these basic metrics, and less on maximizing realism.

Privacy Protection. Another key metric for de-identification is how well it can prevent re-identification. The examples shown in Figure 4.6 give a qualitative comparison of the different approaches. Focusing on the characteristics humans rely on when recognizing faces (see Section 2.4.2), such as the shape and color of the eyes, the thickness of the lips or the shape of the nose, it can be seen that all anonymized images are significantly altered. DetailedPrivacy can change the color of the eyes (see rows two and four) and

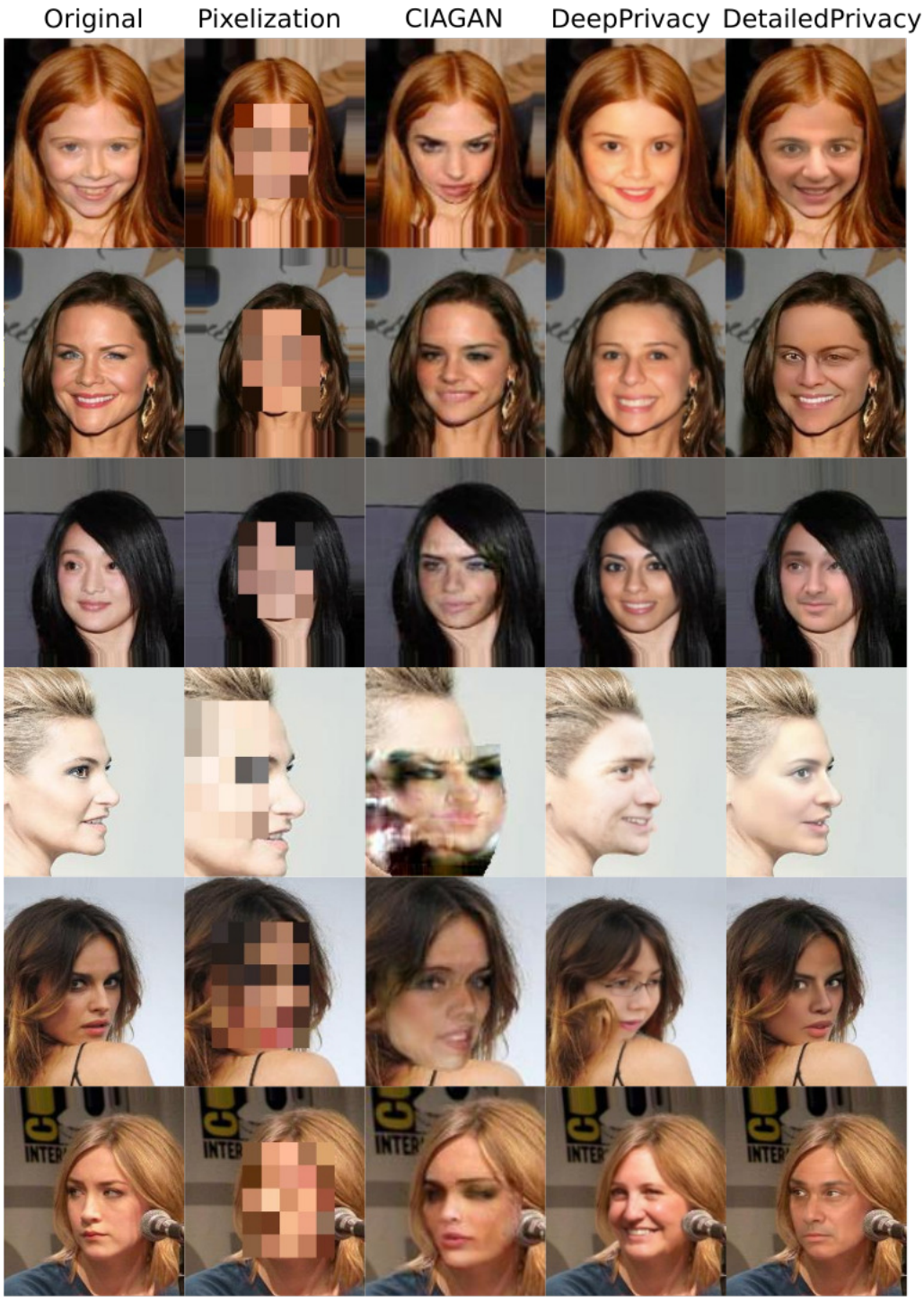


Figure 4.6: Qualitative comparison of images from CelebA (cf. Section 3.2.1) de-identified using pixelization with a 16 by 16 blur kernel, CIAGAN, DeepPrivacy and our approach, DetailedPrivacy.

significantly modify the shape of the nose (see rows one, three and six), while the changes in the form of the lips and eyes are more subtle. For CIAGAN

and DeepPrivacy, on the other hand, these changes are more pronounced. The retention of these characteristics can be traced to the usage of landmarks and keypoints. The sparse keypoints used by DeepPrivacy only retain the distance between the eyes and nose or the approximate width of the face, giving the model the freedom to change facial features. CIAGAN employs more detailed landmarks, but omits the outer landmarks of the mouth region, as well as those around the eyes and the lower part of the nose. Therefore, it leaves their exact shape unspecified. In contrast, DetailedPrivacy uses 98 landmarks that clearly outline many facial features, but still achieves good anonymization by relying on the guidance from the source face to reduce the resemblance to the original. For a quantitative evaluation of the privacy enhancement, we use the TAR measured with ArcFace or FaceNet. As discussed in Section 3.2.2, a lower TAR indicates anonymized faces are harder to re-identify. The results shown in Table 4.1 demonstrate that CIAGAN achieves the strongest anonymization, reducing the TAR from 98.60 % to 3.40 % when FaceNet is used as the recognition tool and from 96.13 % to 5.83 % for ArcFace in our experiments. DeepPrivacy is outperformed by our approach for FaceNet, but achieves better privacy protection on ArcFace. DetailedPrivacy provides reasonable privacy protection, reducing the TAR to 9.03 % for FaceNet and 11.47 % for ArcFace. For most of the results above, FaceNet outperforms ArcFace in terms of re-identifying anonymized faces in our setting. In contrast, the opposite is true for DetailedPrivacy, as its source selection process depends on FaceNet to choose sources that ensure significant differences to the originals. By maximizing the distance between the original and the de-identified images specifically in FaceNet's feature space, this influences privacy protection when performance is measured with the same recognition model. However, our de-identification approach demonstrates comparable protection against ArcFace, which is not part of the anonymization process. Thus, illustrating the strategy's broader effectiveness. Some examples of pairs from LFW successfully anonymized with DetailedPrivacy are given in Figure 4.7. On the other hand, Figure 4.8 shows instances where ArcFace re-identified the anonymized faces. Notably, failed anonymization often occurs repeatedly for the same individual, which might indicate especially distinctive traits (e.g., an unusual head shape) that are difficult to conceal. Furthermore, in these failure cases, the face replacement is

Figure 4.7: Examples of matched pairs from LFW (see Section 3.2.1) that were successfully anonymized with DetailedPrivacy, preventing ArcFace recognition. The left image in each pair is always the one left unchanged, while the right has been anonymized.

Figure 4.8: Examples of matched pairs from LFW (see Section 3.2.1) where DetailedPrivacy anonymization did not prevent ArcFace recognition. The left image in each pair is unchanged, while the right image has been anonymized.

often incomplete, a limitation of our approach we will further discuss in Section 4.2.3. As CNN-based classifiers are often biased towards texture [Geirhos et al., 2019], we speculate that this influences re-identification. Even though DetailedPrivacy is outperformed on this metric by other approaches, we argue that the primary barrier to the widespread adoption of de-identified data is not insufficient privacy protection but rather limited data utility retention. While all approaches can significantly enhance privacy, they will only be widely adopted if they are suitable for the intended task, in a way comparable to the original data.

Retention of Detailed Face Data Utility. Therefore, assuming a reasonable level of privacy protection, from our point of view, the primary focus should be on metrics that directly assess data utility retention. To this end, we use the retention of head pose and facial expression to estimate the utility for downstream tasks linked to the analysis of humans. Our evaluation of pose retention given in Table 4.2 demonstrates that our approach maintains the yaw and roll Euler angles better than all other approaches. However, as the example images in Figure 4.9 show, DetailedPrivacy struggles to create high-quality faces for high pitch angles. Overall, DeepPrivacy achieves a better result on the Mean Absolute Error across the three Euler angles because it performs much better on pitch. The poor performance of CIAGAN in this task is likely a consequence of its training on CelebA, which has a strong bias towards frontal images, rather than a fundamental limitation of the model. Consequently, it tends to generate unrealistic frontalized faces for extreme poses (see, e.g., Figure 4.6 row four). This results in bad retention of the Euler angles. DetailedPrivacy is very effective at preserving facial expression,

De-ID Method	Yaw (↓)	Pitch (↓)	Roll (↓)	MAE (↓)
Original	4.42	6.67	3.25	4.81
CIAGAN	8.31	10.03	5.22	7.85
DeepPrivacy	4.31	6.63	3.52	4.82
DetailedPrivacy	4.16	8.14	3.03	5.11

Table 4.2: Yaw, pitch, roll, and Mean Absolute Error (MAE) of Euler angles, as computed using Hopenet [Ruiz et al., 2018] on the Biwi dataset (see Section 3.2.1). "Original" denotes the results on the unaltered dataset, while "CIAGAN," "DeepPrivacy," and "DetailedPrivacy" show performance after the respective de-identification method was applied.

Figure 4.9: Two examples from the Biwi dataset (cf. Section 3.2.1) for two different volunteers. For each, the top images are the originals, while the bottom images were anonymized using DetailedPrivacy. The approach is robust to large yaw and roll angles, but often creates unrealistic images for high pitch angles (see image six of the first example and image three of the second).

as can be seen in the examples given in Figure 4.10, even though there is some complexity to human emotion that cannot even be captured with the 98 landmarks guiding our approach. Table 4.3 demonstrates that it surpasses all other de-identification methods on emotion retention.

The baseline accuracy for emotion classification on original JAFFE (see Section 3.2.1) images using DeepFace's emotion recognition [Serengil and Ozpinar, 2021] is 66 %. When the images are de-identified with CIAGAN and DeepPrivacy, this number drops to 27 % and 29 %, respectively. DetailedPrivacy, on the other hand, retains 57 % accuracy. DeepPrivacy's poor

De-ID Method	Accuracy JAFFE (↑) [%]	Accuracy AffectNet (↑) [%]
Original	66	44
CIAGAN	27	20
DeepPrivacy	29	N/A
DetailedPrivacy	57	31

Table 4.3: Evaluation of performance retention for emotion detection after de-identification on the JAFFE and AffectNet datasets (see Section 3.2.1). First, the accuracy is measured on the original images and then again after CIAGAN, DeepPrivacy or our approach was applied.

Figure 4.10: Comparison of images from JAFFE (left) and AffectNet (right). See Section 3.2.1 for details. The top row consists of original unaltered images from the respective dataset, while the images in the bottom row were anonymized with DetailedPrivacy.

performance is expected, as the model never sees the original image and only takes seven key points (left/right eye, left/right ear, left/right shoulder, nose) to guide the anonymization process, leaving the expression ambiguous. Thus, it creates a random facial expression, with a bias to those appearing most often in its training dataset (FDF [Hukkelås et al., 2019])
CIAGAN, despite utilizing a subset of 27 out of the 68 facial landmarks that are extracted with dlib [King, 2009], performs even worse on this task. This is likely due to the absence of the outer landmarks of the mouth region and the landmarks around the eyes and the lower part of the nose, which means that the expression is only coarsely defined. Moreover, the model struggles with faces viewed at larger angles, where lower image quality further impacts its effectiveness. On the other hand, as DetailedPrivacy utilizes 98 landmarks clearly outlining detailed facial expressions, it achieves the best results on this metric.
As the JAFFE dataset is rather limited in size, we repeat our experiments on the larger AffectNet (see Section 3.2.1) using ResidualMaskingNetwork [Pham et al., 2021], because the classes predicted by this model align with the ground truth given by the new dataset. The results demonstrate that our approach preserves expressions on this dataset as well, though accuracy drops from 44 % for the baseline to 31 % after anonymization (cf. Table 4.3). As DeepPrivacy cannot be fairly evaluated on this dataset we do not include its results. This is because AffectNet only provides tightly framed facial images,

lacking the keypoints at the shoulders and the background surrounding the face, which DeepPrivacy relies upon for quality. CIAGAN's performance is probably negatively affected because AffectNet includes more faces at larger angles than JAFFE.

4.2.2 Ablation Study

In this section, we validate the suitability of the source selection strategy proposed for DetailedPrivacy. Table 4.4 presents the effect of several alternative ways to choose the source image on privacy protection against FaceNet and ArcFace, image quality, as well as retention of expression and pose.
Our results demonstrate that randomly selecting a source from the library of synthetic images significantly lowers privacy protection, reduces performance on emotion detection and slightly decreases pose retention. However, image quality measured by KID and FID remains roughly the same.
Similarly, choosing a source image solely based on maximal distance from the target in the face embedding space without accounting for pose similarity also results in weaker privacy protection compared to the proposed approach. Surprisingly, selecting the closest sources enhances emotion detection performance after de-identification. This likely occurs due to expression features being tightly coupled with identity features in the face similarity embedding. Therefore, faces that already have expressions resembling the original will be favored when selecting the closest source faces. This facilitates the exact replication of the expression by FSGANv2, leading to better results. Nevertheless, this method fails as a de-identification strategy because it leads to an

De-ID Method	FaceNet (↓) [%]	ArcFace (↓) [%]	KID (↓)	FID (↓)	Acc. JAFFE (↑) [%]	MAE Pose (↓)
DetailedPrivacy	9.03 ± 1.01	11.47 ± 2.25	0.0146 ± 0.0008	13.26	57	5.11
Random	21.87 ± 3.10	21.53 ± 3.24	0.0141 ± 0.0008	13.05	52	5.29
Furthest	11.00 ± 3.09	16.03 ± 3.39	0.0149 ± 0.0009	13.71	56	5.33
Closest	50.10 ± 3.11	42.53 ± 3.87	0.0114 ± 0.0008	11.04	61	5.26

Table 4.4: Comparison of several alternative ways to select source faces for de-identification with DetailedPrivacy. The first row gives the baseline results. "Random" indicates that images were selected randomly from the source library, whereas "Furthest" and "Closest" denote choosing the images with the largest or smallest distance to the target in the facial similarity embedding space without accounting for pose similarity. The accuracy of emotion detection after de-identification (JAFFE dataset) and the Mean Absolute Error of pose estimation (Biwi dataset) are determined after de-identification.

unacceptable re-identification rate of 50.10 % for FaceNet.

4.2.3 Limitations

Even though DetailedPrivacy demonstrates robust data utility retention and privacy protection for faces with challenging poses, expressions and backgrounds, the image quality of the de-identified images suffers for unusual poses. An example is shown in the first image of the first row of Figure 4.11, but the same phenomenon can be observed in Figure 4.9, especially for large pitch angles of the face as discussed in Section 4.2.1. The underlying cause for this limitation is the lack of images of similar poses in the library of synthetic source faces, as FSGANv2 depends upon such images as a starting point for its iterative adjustment of the source to the target position. The greater the difference between the facial yaw, pitch, or roll angles, the greater the degradation of identity and texture quality [Nirkin et al., 2022]. In our pipeline, this could be mitigated by including faces at more unusual angles in the source library. Thus, using the source selection process, faces closer to the target

Figure 4.11: Failure cases of images from CelebA (see Section 3.2.1) anonymized with DetailedPrivacy, highlighting the limitations of our approach. Large head angles often lead to reduced image quality, while significant occlusions can be partially removed or result in unnatural skin tones in the synthesized faces. Additionally, generated elements such as teeth and glasses can appear artificial. In some instances, the surrogate face fails to fully cover the target face, leading to incomplete anonymization.

pose could be chosen. However, we have only included faces with yaw angles up to ± 25 °. Generating faces with even larger angles with the technique used in this work [Meißner et al., 2022] leads to images with more artifacts and can cause significant changes in identity features, causing inconsistency between different views.

Another limitation rooted in the source library is the unrealistic appearance of generated glasses, which are often incomplete, as shown in the second and third images of the first row of Figure 4.11. This could be addressed by removing people with glasses from the source library, but this would also reduce the diversity of the anonymized images. Synthesis of teeth also often leads to artificial results, which can be seen in the second to last image.

While our approach deals well with small occlusions, larger occlusions from sunglasses, microphones (last two images, first row) or hands covering the face (first image, second row), are challenging to recreate in a realistic fashion. Instead, they are often partly removed. Additionally, the second and third images in the second row show that unrealistic skin tones can be generated for people with beards or other large facial occlusions.

Finally, the last image shows that faces are sometimes only incompletely covered by the surrogate face, which could negatively affect privacy protection.

4.3 Evaluation of StablePrivacy for Anonymizing Training Data

While DetailedPrivacy was designed to retain fine-grained facial details, StablePrivacy focuses on retaining coarser features. It can create high-quality anonymizations even for challenging images, making it ideal for de-identifying training data for tasks such as face detection. In this section, we compare our approach to the state of the art for this use case.

4.3.1 Privacy Protection and Image Quality

Similarly to the last section, we first evaluate privacy protection and image quality of StablePrivacy, before discussing task-specific data utility retention. The example faces anonymized with the different approaches shown in Figure 4.12 illustrate these properties qualitatively. Our anonymizations look highly realistic and results are obtained robustly for challenging images containing occlusions such as hats (row two), beards (row four) or glasses (row five) as well as difficult poses (row two) and expressions (row seven). Moreover, as it does not use keypoints, our approach is free to change features that make faces similar in human perception, such as the shape of the eyes or the thickness of the lips and eyebrows (cf. Section 2.4.2). Notably, in several examples the shape of the entire face is altered (e.g., rows six and seven).
Table 4.5 includes the results from the last section for original unaltered data,

De-ID Method	FaceNet (↓) [%]	ArcFace (↓) [%]	KID (↓)	FID (↓)
Original	98.60 ± 0.76	96.13 ± 1.81	N/A	N/A
Face Pixelization 16 × 16	0.56 ± 1.67	0.33 ± 0.26	0.0417 ± 0.0012	43.09
CIAGAN	3.40 ± 0.65	5.83 ± 1.97	0.0105 ± 0.0007	13.30
DeepPrivacy	10.90 ± 1.93	6.63 ± 2.12	0.0014 ± 0.0002	2.37
DetailedPrivacy	9.03 ± 1.01	11.47 ± 2.25	0.0146 ± 0.0008	13.26
DeepPrivacy2	11.64 ± 1.97	8.60 ± 1.76	**0.0004 ± 0.0002**	**1.34**
LDFA	12.92 ± 2.51	9.40 ± 2.39	0.0014 ± 0.0002	2.58
StablePrivacy	**0.70 ± 0.48**	**0.90 ± 0.33**	0.0017 ± 0.0003	3.36

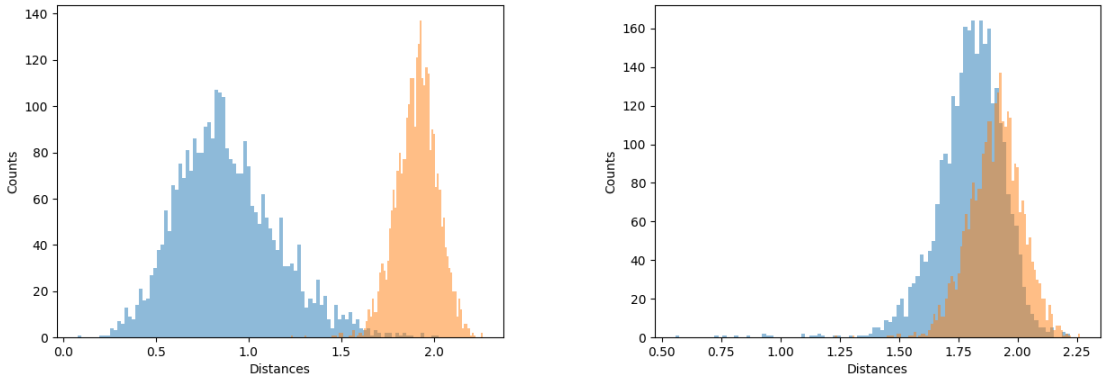
Table 4.5: Comparison of privacy protection concerning FaceNet or ArcFace and image quality (KID, FID) evaluated on LFW. The baselines (original and pixelization) show the upper and lower limits for re-identification. StablePrivacy offers the best privacy protection among the synthesis-based face de-identification approaches.

Figure 4.12: Visual comparison of StablePrivacy with other de-identification approaches when applied to images from LFW (see Section 3.2.1). Our anonymizations are highly realistic even for challenging images and substantially alter the face providing good privacy protection.

anonymization by pixelization as a baseline for traditional methods without consideration of utility, and the keypoints-based de-identification methods (CIAGAN, DeepPrivacy and DetailedPrivacy) for comparison. Additionally, we evaluate our approach against the recent keypoints-free approaches LDFA and DeepPrivacy2 (see Section 3.2.3). While DeepPrivacy, DeepPrivacy2, DetailedPrivacy and LDFA all substantially reduce the probability of re-identification, there remains a considerable risk which might be unacceptable for some applications. In comparison, CIAGAN offers much stronger protection, but reduces the image quality significantly.

Our method, StablePrivacy, robustly creates images of a reasonable quality, while substantially improving upon all others (besides pixelization) concerning privacy protection for both FaceNet (0.70 %) and ArcFace (0.90 %). As discussed in Section 4.2, results for FaceNet are better as the model is utilized in the source selection process, but the performance for ArcFace is comparable.

The effectiveness of privacy protection is further illustrated by Figure 4.13, showing distance distributions for mismatched (orange) and matched (blue) LFW pairs before (Figure 4.13a) and after (Figure 4.13b) anonymization of matched pairs with our approach. As mismatched pairs are not de-identified, their distribution is unaffected. At the same time, the distances in matched pairs are increased significantly as one of the images is anonymized, causing distributions to substantially overlap. Thus, it is not possible to select a threshold value that unambiguously separates them, which leads to a low re-identification rate as explained in Section 3.2.2. All pairs that could nevertheless be classified as matching pairs after de-identification with StablePrivacy are shown in Figure 4.14. Even though privacy protection has



(a) Distances between the pairs in the LFW evaluation protocol with original, not anonymized data.

(b) Distances between the pairs in the LFW evaluation protocol with one of the images of each matched pair anonymized by StablePrivacy.

Figure 4.13: Comparison of distances in the face similarity space of ArcFace of matched pairs (blue) and mismatched pairs (orange) from the LFW dataset. Matched pairs originally show the same person and have low distances, while mismatched pairs show different people and therefore have high distances. After anonymization of the matched pairs, their distribution is more aligned with that of mismatched pairs, demonstrating the strong privacy protection of StablePrivacy.

Figure 4.14: All image pairs from the LFW dataset (see Section 3.2.1) for which anonymization with StablePrivacy has failed. Many of them look significantly different to human observers, which leads us to the assumption that features outside the face, such as hair, could affect re-identification among the limited number of identities in LFW.

failed in those cases, many of the pairs look visibly different to a human observer. One explanation for the result could be that the face recognition tool is exploiting features that are not relevant to the human visual system. Another explanation could be identification based on features outside the face, such as hair, which remain unchanged by StablePrivacy when used with our default parameters. Although such features would probably not suffice to uniquely identify a person among a larger population, this is more likely within the rather small LFW dataset. The idea that features outside the face influence recognition on this benchmark is supported by our experiments discussed later in this chapter (see Section 4.3.4) showing that altering a larger area around the face with inpainting somewhat improves de-identification.

4.3.2 Utility Retention for Training Face Detection Models

The previous experiments have demonstrated that our approach can produce high-quality realistic images and provide state-of-the-art privacy protection. In this section, we present our experiments illustrating StablePrivacy's ability to retain utility for training face detection models on anonymized data.
As discussed in Section 3.2.2, we compare the mean average precision (mAP) of a DSFD face detection model [Li et al., 2019] trained on the original WIDER FACE dataset with that of the same model trained on this dataset anonymized with one of the different de-identification approaches. To perform well on this task, these approaches must create surrogates that resemble real faces so that the detection model can transfer its ability to localize faces learned on the de-identified data to detecting real faces at evaluation time. While detailed features like expression and facial pose do not need to be retained for individual images, preserving the diversity found in the original dataset is important so the face detector can learn from a realistic distribution. As the images in the dataset typically contain multiple faces, each needs to be anonymized individually. We described how this is implemented for StablePrivacy in Section 4.1.2. CIAGAN has been adjusted to apply a similar process, using the ground truth locations provided by WIDER FACE to cut out areas of the image containing faces to anonymize and then insert them back into the original image one by one. LDFA, DeepPrivacy and DeepPrivacy2 have their own processes to anonymize multiple faces as described in their

respective publications [Hukkelås and Lindseth, 2023; Klemp et al., 2023]. Still, to ensure a fair comparison, we adjusted their de-identification pipelines to use ground-truth-provided bounding box coordinates to outline the faces to anonymize instead of bounding boxes predicted by an additional detection model, as suggested by Klein [2023]. DetailedPrivacy cannot be included in this evaluation because it requires the extraction of detailed landmarks, which is often not possible on images of this dataset.

Following Klomp et al. [2021], we trained the DSFD model for 60,000 iterations using the official PyTorch implementation[3] with a VGG-16 backbone (see Section 2.2), a learning rate of 2.5×10^{-4} and a batch size of four. We still measure systematically lower detection performances, indicating differences in the experimental setup. According to our results presented in Figure 4.15, CIAGAN has the weakest utility retention, likely because its required landmarks are difficult to extract reliably for small faces and its limited ability to create high-quality outputs for occluded and non-frontalized faces. In contrast, DeepPrivacy achieves much better results as it only needs seven keypoints from the nose, eyes, ears and shoulders, which can be acquired even for more challenging images. Notably, the detection model trained on its anonymized

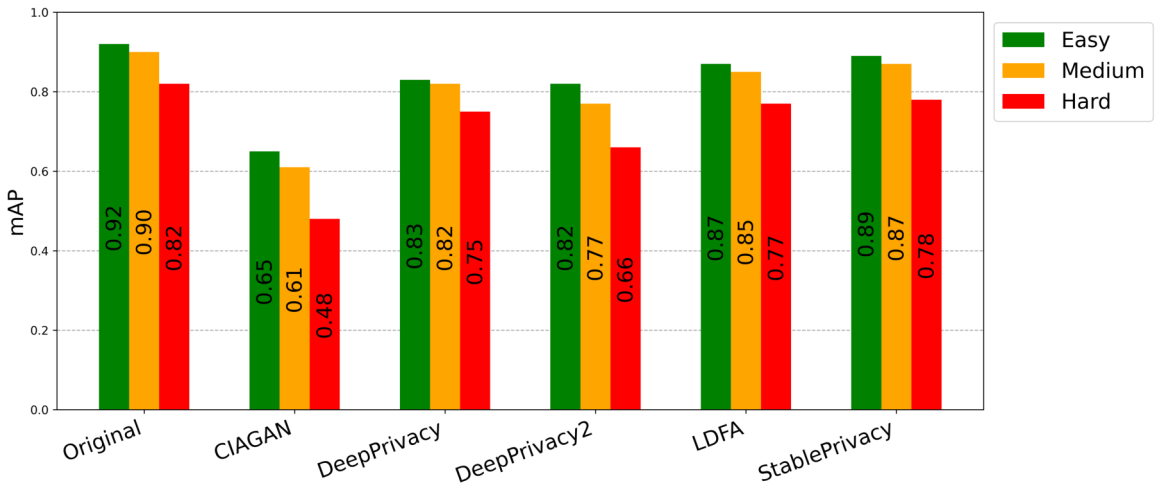


Figure 4.15: Results of DSFD face detection trained on the original and on anonymized WIDER FACE datasets, revealing the impact of various anonymization techniques. Among them, StablePrivacy delivers a notable performance advantage compared to alternative de-identification approaches.

[3] https://github.com/Tencent/FaceDetection-DSFD

data even outperforms that of DeepPrivacy2 by 1 % on easy faces and by an even more substantial margin on medium (5 %) and hard (9 %) faces. However, LDFA, the second best approach in our evaluation, achieves even better results with mAP on average across all levels of difficulty being only 5 % worse then when DSFD is trained on the original data. Still, StablePrivacy performs best, improving upon LDFA by 2 % for the easy and medium level and 1 % for the hard level. Additionally, StablePrivacy offers much stronger privacy protection as demonstrated in the previous experiments. Nevertheless, there is still a performance gap of 3 to 4 % compared to training the model on original data. To confirm that findings on utility retention are not specific to the chosen detection model, we replace DSFD with YOLOv8 nano [Redmon et al., 2015; Jocher et al., 2023], a more recently developed model and repeat the experiment. As can be seen in Figure 4.16, the results remain comparable. As before, we use the original training split of WIDER FACE and the same dataset anonymized by the different de-identification approaches. We train the YOLO model for 90 epochs with a batch size of 16. Although absolute results differ compared to DSFD, with slightly better mAP for most approaches for the easy and medium difficulty levels and worse for the hard level, importantly, the relative performance ranking of the approaches remains the same.

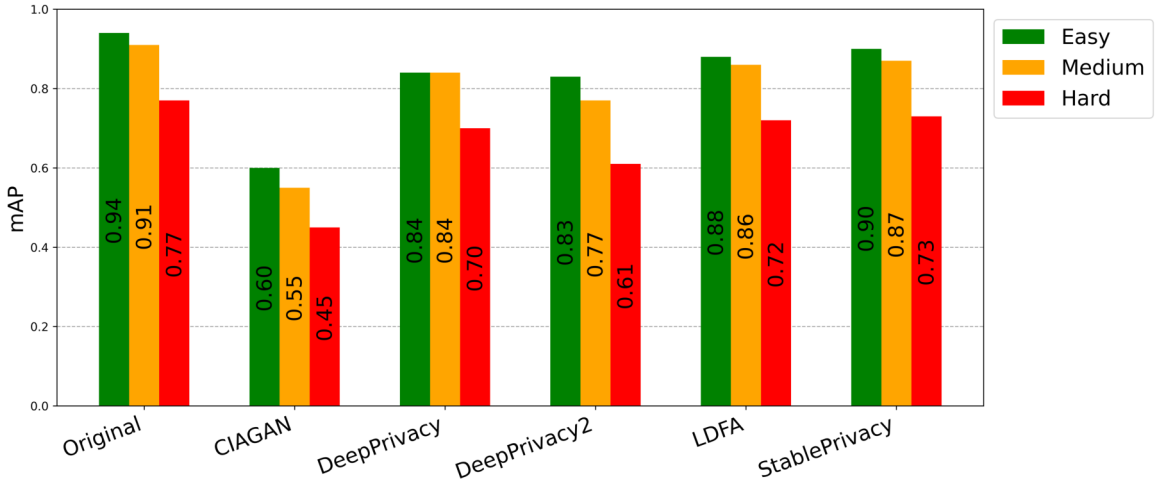
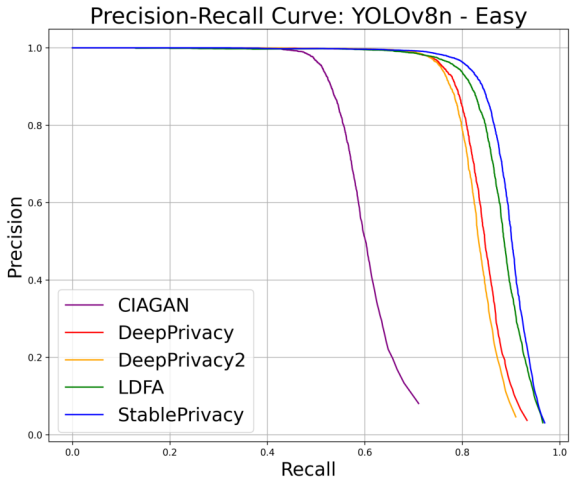
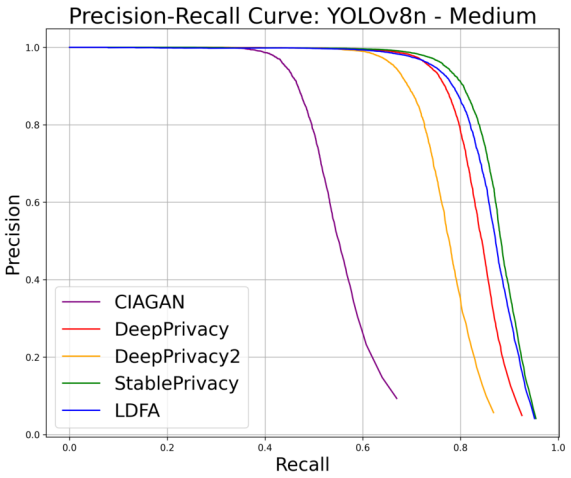


Figure 4.16: Results for a YOLOv8 nano face detection model trained on the original and on anonymized WIDER FACE datasets, showing the influence of various anonymization techniques. Similarly to the DSFD model, the results demonstrate that our approach, StablePrivacy, performs best for retaining utility for training deep-learning-based face detection models on anonymized data.

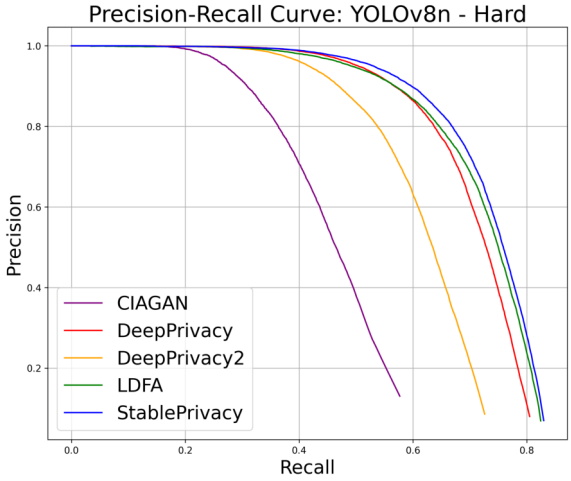
Figure 4.17 illustrates the precision-recall curves from which these results are calculated as described in Section 3.2.2. StablePrivacy achieves better precision for all corresponding recall values, even though its curve converges with LDFA's towards final segment. The good performance of StablePrivacy is probably due to its ability to incorporate structural guidance, allowing it to create realistic images even for challenging scenes as shown in Figure 4.18. These examples illustrate that our approach can create good results for large groups (top right and directly below), heavy occlusions (the hat in the upper left image, the skier's helmet, and the instruments in the bottom right image) and challenging light conditions (see the kayaker).



(a) YOLOv8n for difficulty level easy.



(b) YOLOv8n for difficulty level medium.



(c) YOLOv8n for difficulty level hard.

Figure 4.17: The precision-recall curves from which the mAP for face detection is calculated. Comparing the models trained on data de-identified by the different approaches, it can be seen that StablePrivacy has the highest precision for all recall values.

Figure 4.18: Images from WIDER FACE (see Section 3.2.1) de-identified by StablePrivacy. The anonymized faces look realistic, even for challenging images containing large groups (top right and directly below), heavy occlusions such as the hat in the first image, the skier's helmet or the instruments in the bottom right image and difficult light conditions as in the image of the kayaker.

4.3.3 Visualizing the Influence of Anonymization on Detection

In this section, we aim to validate whether the detection model utilizes the facial features available when faces are anonymized with surrogates and does not rely solely on cues from other regions such as hair or the body. Addition-

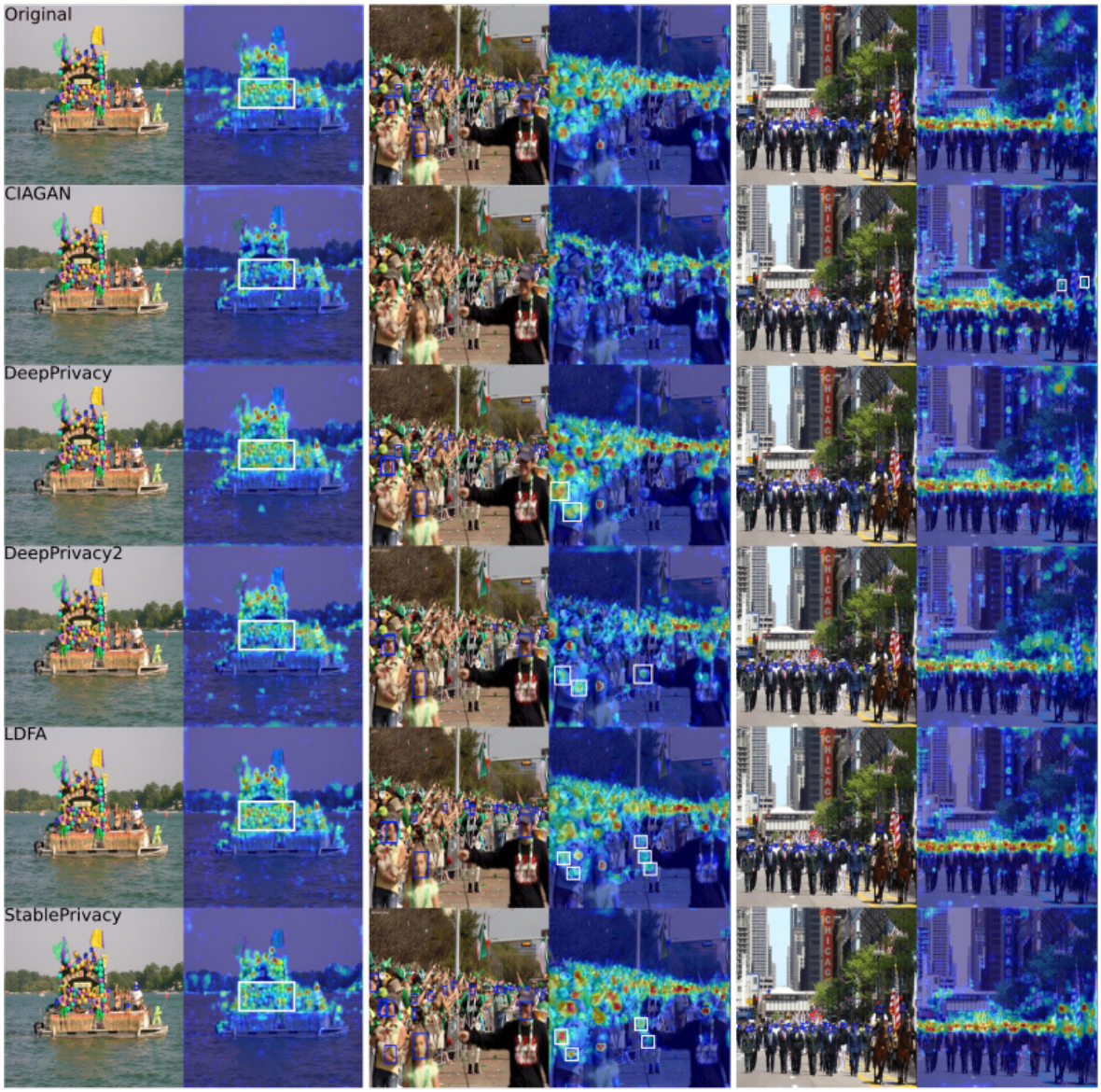


Figure 4.19: The left images of each column show bounding boxes predicted by YOLOv8 trained on original data and data anonymized by the different approaches. The right images depict the corresponding Class Activation Maps (CAMs) using a red-to-blue color scale, with red indicating the areas with the highest influence on the model's prediction. The white boxes highlight regions that demonstrate model behavior discussed in the text. Images are from the original WIDER FACE validation split (see Section 3.2.1).

ally, we examine how detection strategies differ between models trained on the original data compared to those trained on the various anonymized data. To this end, we visualize the class activation maps computed using Eigen-CAM [Muhammad and Yeasin, 2020; Gildenblat, 2021] applied to the last convolutional layer of the classification branch of YOLO's head in Figure 4.19. The results indicate that all synthesis-based anonymization approaches lead to detectors utilizing facial features in a manner similar to those trained on original data. However, CIAGAN-derived YOLO shows a noticeably weaker response to the faces in the images from the first and second column, failing to highlight many relevant areas. In the third column, it performs better, but it still misses the more challenging faces, such as those of the riders (cf. white boxes). Moreover, the first column demonstrates a similar response of all models to round shapes such as the balloons. On the other hand, the second column reveals a divergence: while most models that learned detection on anonymized data react strongly to bare skin (for example, the raised hand of the person in front or the hand of the woman holding the baby on the left), the model trained on the original data shows considerably less sensitivity. This behavior may be attributed to the presence of at least some low-quality faces in anonymized datasets, where unrealistic anatomical characteristics drive detectors to focus more on color cues.

4.3.4 Influence of Parameters on the Privacy Utility Trade-Off

As explained in Section 4.1.2, StablePrivacy has multiple parameters affecting the trade-off between privacy and data utility of the output data. Here, we analyze their exact impact.

Strength Parameter. We begin by examining how different values of the strength parameter affect performance, as summarized in Figure 4.20. To reduce variability, we retain a single source image across all targets. As the strength value grows from 0.1 to 1, the model receives progressively weaker structural guidance from the original image, causing image quality to decrease, as shown by KID and FID rising from 0.0005 and 1.35 to 0.0115 and 10.89, respectively. Concurrently, privacy protection continually improves, as indicated by the diminishing TAR when tested on LFW with FaceNet or ArcFace. Although a low strength of 0.1 only slightly reduces TAR compared

Strength	FaceNet (↓) [%]	ArcFace (↓) [%]	KID (↓)	FID (↓)
0.0 (original)	98.60 ± 0.76	96.13 ± 1.81	N/A	N/A
0.1	93.80 ± 1.33	90.23 ± 2.12	0.0005 ± 0.0002	1.35
0.2	72.17 ± 2.19	73.17 ± 2.85	0.0007 ± 0.0002	1.61
0.3	39.57 ± 2.85	42.67 ± 4.15	0.0010 ± 0.0002	2.01
0.4	15.53 ± 2.31	18.23 ± 2.73	0.0017 ± 0.0003	2.67
0.5	5.23 ± 1.44	6.13 ± 1.48	0.0028 ± 0.0004	3.69
0.6	2.43 ± 0.73	2.22 ± 0.73	0.0045 ± 0.0005	5.06
0.7	**1.47 ± 0.60**	**1.00 ± 0.60**	**0.0063 ± 0.0005**	**6.58**
0.8	1.33 ± 0.42	0.67 ± 0.21	0.0078 ± 0.0007	7.94
0.9	1.07 ± 0.36	0.50 ± 0.27	0.0092 ± 0.0008	9.10
1.0	0.77 ± 0.40	0.47 ± 0.16	0.0115 ± 0.0008	10.89

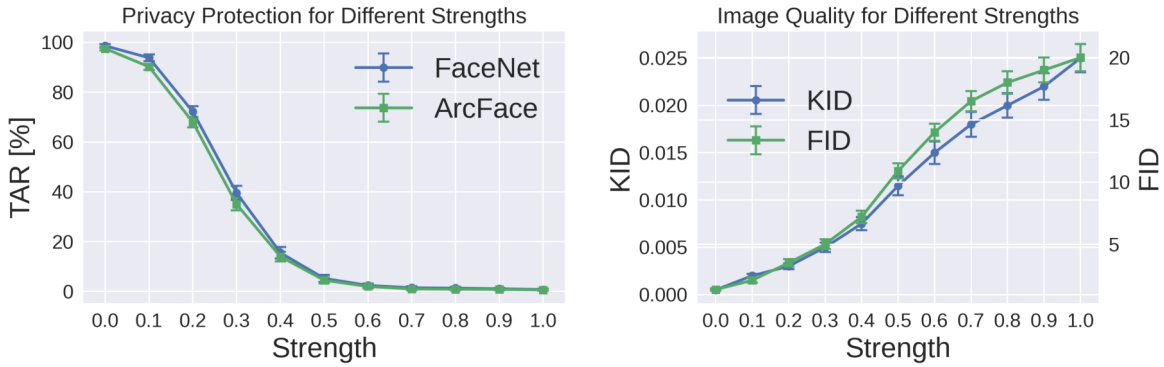


Figure 4.20: The influence of the strength parameter on privacy protection and image quality of StablePrivacy. Larger strength results in better de-identification, but reduces image quality. In our main experiments, we use a value of 0.7.

to the unprotected original data from 98.60 to 93.80 (FaceNet), it declines more steeply between 0.1 and 0.5, reaching 5.23 %. Further increases in strength produce diminishing returns on improving privacy protection. The images in Figure 4.21 illustrate this trend. For instance, looking at the first row, faces generated using a low strength value look highly realistic but resemble the original. On the other hand, for a strength of 0.9, faces look completely altered but contain obvious artifacts such as the unrealistic skin tone in the first row or the hair in the image in the last row. Considering the trade-off between privacy and quality, we choose a strength value of 0.7 for our primary experiments, which sufficiently obfuscates identities without excessively compromising visual quality.

Figure 4.21: Example images from LFW (cf. Section 3.2.1) anonymized with our approach for different choices of the strength parameter (0.1 to 0.9), illustrating the trade-off between privacy protection and realism. For low strength values, faces clearly resemble the original, while for high values, they appear completely altered but contain more obvious artifacts, such as the hair in the last image in the last row.

CFG Scale. The next crucial parameter affecting the balance between privacy and utility in anonymized images is the CFG (guidance) scale [Ho and Salimans, 2021]. The higher it is, the more closely the Stable Diffusion model follows the image prompt. This becomes apparent in Figure 4.22. While for the example in the first row facial features such as the thickness of the lips and the shape of the nose change more significantly for higher values of the CFG scale, the face is altered to a slightly more frontalized position. The deviation from the original towards the image prompt is even more evident in the second row, with the increasing change in expression and the removal of the slight occlusion by the snowflakes. This could reduce the diversity of expression, pose, occlusions or other features in the anonymized training dataset by aligning it with the much smaller source library, potentially

Figure 4.22: Example images from the LFW dataset (see Section 3.2.1) anonymized with StablePrivacy. The higher the CFG (guidance) scale, the more closely the Stable Diffusion model follows the image prompt during inpainting. Facial features such as the thickness of the lips and the shape of the nose resemble the source more closely for higher values of the CFG scale. At the same time, pose and expression also align more closely with the image prompt, potentially reducing the diversity of anonymized training datasets.

Guidance Scale	FaceNet (↓) [%]	ArcFace (↓) [%]	KID (↓)	FID (↓)
1.0	13.00 ± 1.32	14.07 ± 1.79	0.0009 ± 0.0002	2.18
3.0	2.03 ± 0.43	3.07 ± 0.73	0.0012 ± 0.0002	2.75
5.0	**0.70 ± 0.48**	**1.17 ± 0.47**	**0.0017 ± 0.0003**	**3.38**
7.0	0.53 ± 0.31	0.76 ± 0.47	0.0024 ± 0.0004	4.16

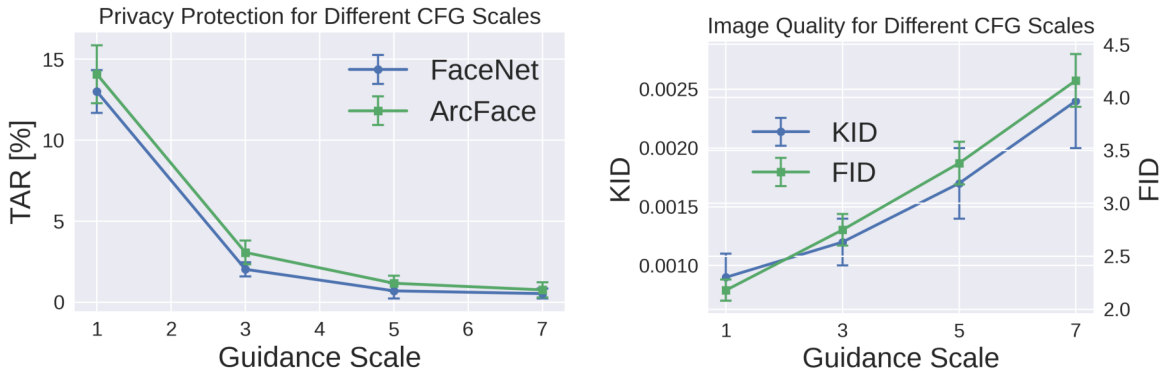


Figure 4.23: The influence of the CFG (guidance) scale on our approach. Larger guidance scales improve privacy protection but reduce image quality. We choose a value of 5.0 for our main experiments, balancing privacy protection and image quality.

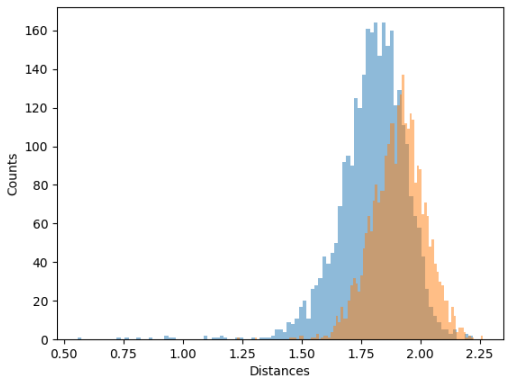
hindering a downstream model from learning the real variety.
Quantitatively, this is confirmed by Figure 4.23. Raising the CFG scale enhances privacy protection as the TAR measured by FaceNet and ArcFace declines from 13.00 % and 14.07 % to 0.53 % and 0.76 %, respectively. At the same time, image quality is diminished with FID increasing from 2.18 to 4.16. As the model adheres more closely to the features of the image prompt, it must deviate more from the original face, thereby anonymization is strengthened. However, this also means that the model has less freedom to incorporate the structural guidance from the SDEdit technique, resulting in lower image quality. For our main experiments, we use a CFG scale of 5.0, since further increases offer minimal benefits to privacy, but substantially reduce visual quality.

Area of Inpainting. Finally, we investigate the effect of increasing the area of inpainting. While enlarging this area to include, for example, hair and ears can improve privacy, it reduces the faithfulness to the original image content, as a larger region is changed. The effect can be seen in the rightmost image of Figure 4.24, showing that objects unrelated to identity, like the medal in this example, can get significantly distorted if they lie within this region. The red solid box in the leftmost image highlights the area of inpainting used by our baseline implementation of StablePrivacy for our experiments on the LFW dataset. It is based on the tight bounding box provided by

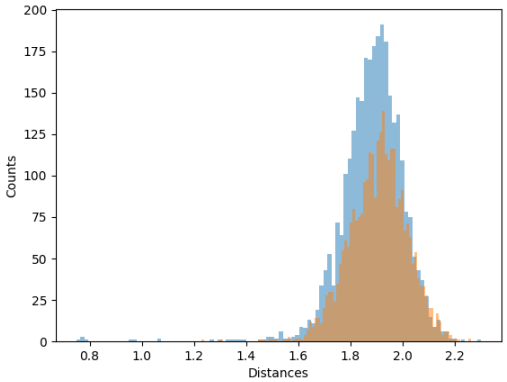
Figure 4.24: Example image from LFW (see Section 3.2.1) showcasing the effect of increasing the area of inpainting during anonymization with StablePrivacy by 15 or 30 pixels in all directions. The red solid bounding box on the original face highlights the default area of inpainting; the dashed bounding box illustrates the extended area. Choosing this parameter too large can arbitrarily change the context of the scene.

Additional Area	FaceNet (↓) [%]	ArcFace (↓) [%]	KID (↓)	FID (↓)
0	**0.70 ± 0.48**	**0.90 ± 0.33**	**0.0017 ± 0.0003**	**3.36**
15 pixels	0.6 ± 0.20	0.53 ± 0.34	0.0040 ± 0.0005	5.44
30 pixels	0.47 ± 0.27	0.33 ± 0.21	0.0052 ± 0.0006	7.08

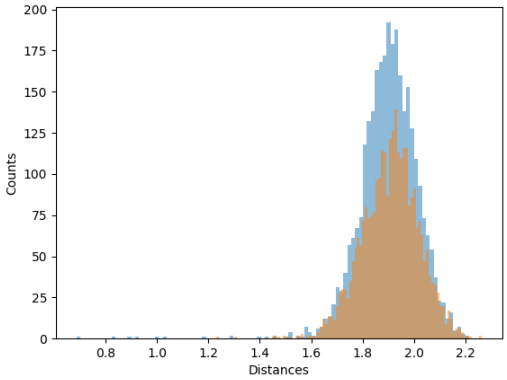
Table 4.6: The influence of the area of inpainting. A larger area improves privacy protection, but reduces image quality. 0 is the default value for StablePrivacy used in all other experiments.



(a) Additional inpainting area = 0



(b) Additional inpainting area = 15 pixels



(c) Additional inpainting area = 30 pixels

Figure 4.25: Comparison of face similarity distances of matched pairs with one image anonymized by StablePrivacy (blue) and mismatched pairs (orange) from LFW for different sizes of the inpainting area. For larger inpainting areas, the distribution of anonymized pairs aligns even more closely with the distribution of pairs showing entirely different people.

DSFD with no additional padding. Extending this area by 15 or 30 pixels to all sides, as illustrated by the dashed bounding box, further improves de-identification to 0.6 and 0.47, respectively, as given in Table 4.6. However, at the same time, FID rises from 3.36 to 7.08, indicating a worse image quality. Figure 4.25 illustrates the effect on the distribution of distances in ArcFace's

embedding space, demonstrating that the distances between mismatched pairs in LFW become indistinguishable from the distance between matched pairs where one image is anonymized with StablePrivacy when the inpainting area is increased. This supports our hypothesis that non-anonymized features outside the face contribute to the re-identification of anonymized faces, as discussed for Figure 4.14 in Section 4.3.2. Nevertheless, we ultimately discard this option as it leaves a large area around the original face to be changed arbitrarily, which potentially leads to reduced data utility.

4.3.5 Privacy Protection for Small Faces

In Section 4.1.2 we proposed using lower values for the strength parameter of StablePrivacy if faces are sufficiently small. Here, we experimentally validate this choice and suggest additional strategies depending on privacy requirements.

The idea is based on the observation that smaller faces are inherently more difficult to re-identify [Knoche et al., 2022]. This allows us to lower our method's strength parameter, improving image quality and potentially data utility without sacrificing privacy. To experimentally find appropriate values for a given size, we simulate the effect of face size by rescaling the images of LFW, then applying our anonymization, and repeating the measurement of the TAR with FaceNet similarly to our experiments in Section 4.3.1. The outcomes in Table 4.7 show that a strength setting of 0.5 offers acceptable anonymity for faces of about 30×30 pixels, but privacy protection deteriorates quickly for larger faces at this strength. Consequently, our primary strategy, outlined in Section 4.1.2, uses a strength of 0.7 for faces exceeding 30×30 pixels, and 0.5 for all other faces. Apart from achieving better data utility, the main advantage of this approach is a significant speed up in computation

Size	Strength = 0.5	Strength = 0.4	Strength = 0.3
30×30 pixels	3.53 ± 0.87	10.13 ± 1.11	28.04 ± 2.38
20×20 pixels	3.46 ± 1.54	8.06 ± 1.41	19.87 ± 2.56
10×10 pixels	2.50 ± 0.95	4.51 ± 1.78	7.37 ± 1.94

Table 4.7: Privacy protection of our approach for different strength values and face sizes measured by TAR with FaceNet.

time as fewer diffusion steps are required for the large majority of the faces in WIDER FACE.
Alternative strategies can be chosen depending on the user's privacy needs. For applications focusing on stringent privacy protection, a uniform strength of 0.7 is recommended. If the priority shifts towards data utility and computational efficiency, scaling the strength with face size becomes viable. To this end, we additionally investigate two alternative scaling schemes:

- **Variant One:** For descending face sizes of 30×30, 20×20, and 8×8 pixels, we employ strength values of 0.5, 0.4, and 0.0, respectively. For larger faces, we use the default value of 0.7.

- **Variant Two:** A more granular approach setting the strength to 0.7 for faces of at least 30×30 pixels, lowering it to 0.5 for faces of at least 25×25 pixels, then to 0.4 for at least 20×20 pixels, to 0.3 for at least 15×15 pixels, to 0.2 for at least 10×10 pixels, and disabling anonymization for all faces smaller than that.

While both scaling schemes slightly outperform our primary strategy in detector accuracy (cf. Table 4.8), they do so only at the expense of privacy protection. The second variant, for example, significantly compromises privacy at smaller face sizes, as indicated by a TAR of 19.87 % for 20×20 pixel faces and a strength of 0.3 (see Table 4.7). In contrast, consistently using a strength of 0.7 for every face size yields the most robust privacy protection among these schemes, but results in slightly worse detector performance and significantly increased computation time.

Size vs. Strength Variants	Easy [mAP] (↑)	Medium [mAP] (↑)	Hard [mAP] (↑)
StablePrivacy	0.89	0.87	0.78
Constant Strength = 0.7	0.88	0.87	0.78
Variant One	0.89	0.88	0.79
Variant Two	0.90	0.88	0.80

Table 4.8: Comparison of the mAP of DSFD when trained on WIDER FACE anonymized by StablePrivacy with different schemes of scaling the strength according to face size.

4.3.6 Impact of Using Anonymized Data on Model Scaling

Thus far, we have shown that using the right parameter choices and strategy to scale strength according to face size, StablePrivacy can create high-quality, realistic-looking anonymized images, provide state-of-the-art privacy protection and preserve utility for downstream training of face detection models.
Here, we extend our assessment of data utility for this task with YOLOv8 using its five different size variants (nano, small, medium, large, and extra-large) to study the impact of increasing the number of model parameters on performance. Our hypothesis is that for larger detection models, the performance gap will widen, as they rely on a sufficient variety of learnable features that may be compromised in anonymized data and have more capacity to overfit on specific artifacts of the anonymized images.
It is validated by Table 4.9, demonstrating that the mean mAP gap across difficulty levels grows from 4.23 % (YOLOv8n) to 5.64 % (YOLOv8x). This trend is evident in Figure 4.26 (a) for the blue line describing StablePrivacy without further modification. Interestingly, when looking more closely at the performance gap for the individual levels of difficulty (cf. Figure 4.26 (b) to (d)), it

Data	Model	Difficulty			Mean ↑ (Δ)
		Easy ↑ (Δ)	Medium ↑ (Δ)	Hard ↑ (Δ)	
Original	YOLOv8n	93.76 (4.20)	91.43 (4.10)	77.07 (4.40)	87.42 (4.23)
	YOLOv8s	95.43 (5.01)	93.66 (4.79)	80.85 (4.79)	89.98 (4.86)
	YOLOv8m	95.99 (5.55)	94.39 (4.98)	82.72 (5.29)	91.03 (5.27)
	YOLOv8l	96.21 (5.67)	94.85 (5.25)	83.67 (5.23)	91.58 (5.38)
	YOLOv8x	96.15 (6.09)	94.83 (5.49)	83.98 (5.33)	**91.65** (5.64)
	Average	95.51 (5.30)	93.83 (4.92)	81.66 (5.01)	90.33 (5.08)
StablePrivacy	YOLOv8n	89.56 (-)	87.33 (-)	72.67 (-)	83.19 (-)
	YOLOv8s	90.42 (-)	88.87 (-)	76.06 (-)	85.12 (-)
	YOLOv8m	90.44 (-)	89.41 (-)	77.43 (-)	85.76 (-)
	YOLOv8l	90.54 (-)	89.60 (-)	78.44 (-)	**86.19** (-)
	YOLOv8x	90.06 (-)	89.34 (-)	78.65 (-)	86.02 (-)
	Average	90.20 (-)	88.91 (-)	76.65 (-)	85.26 (-)

Table 4.9: Comparison of YOLOv8 face detection models' mAP scores when trained on original and on StablePrivacy-anonymized datasets for the different difficulty levels of the WIDER FACE dataset. The values in brackets (Δ) describe the performance difference. The row "Average" gives the average over model sizes, while the "Mean" column presents the average across levels of difficulty.

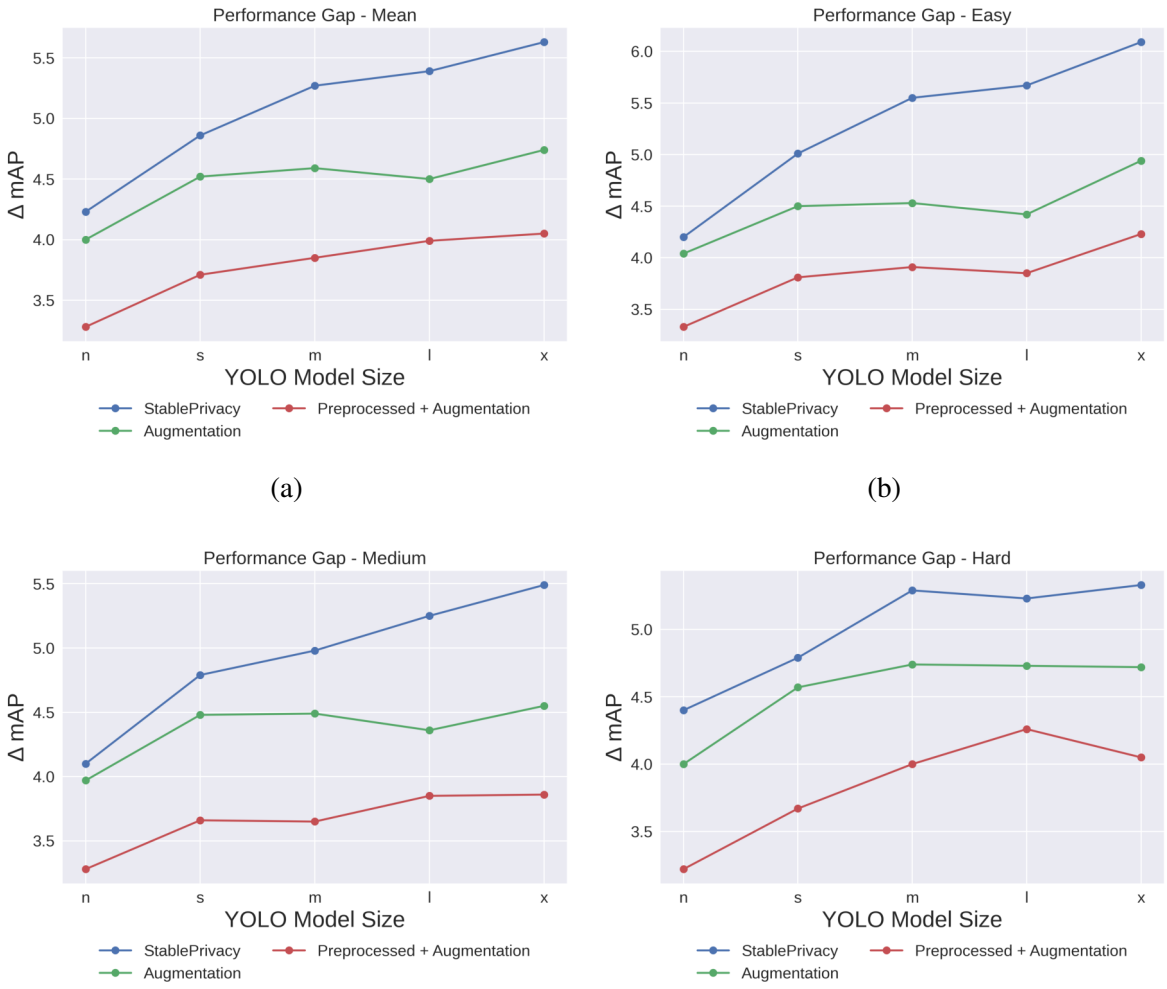


Figure 4.26: The performance gap between the YOLOv8 size variants: nano (n), small (s), medium (m), large (l) and extra-large (x) when trained on original data versus data anonymized by StablePrivacy. The comparison includes three scenarios: (1) StablePrivacy without further modifications, (2) with additional inpainting for data augmentation, and (3) with additional inpainting combined with further preprocessing of the source faces. Among these, the third scenario results in the smallest performance gap.

can be seen that for "hard" the trend differs from the other levels. Particularly, it does not grow significantly after model size m. Concerning the absolute values, Figure 4.27 (a) demonstrates that models based on anonymized data reach their maximum performance at the large scale, while the performance of the model trained on original data improves also beyond that point, though the rate of increase declines. This indicates that the anonymized data imposes constraints on the benefits of scaling model size. Again, the trend is different for the hard level, for which mAP continually improves. We suspect that the

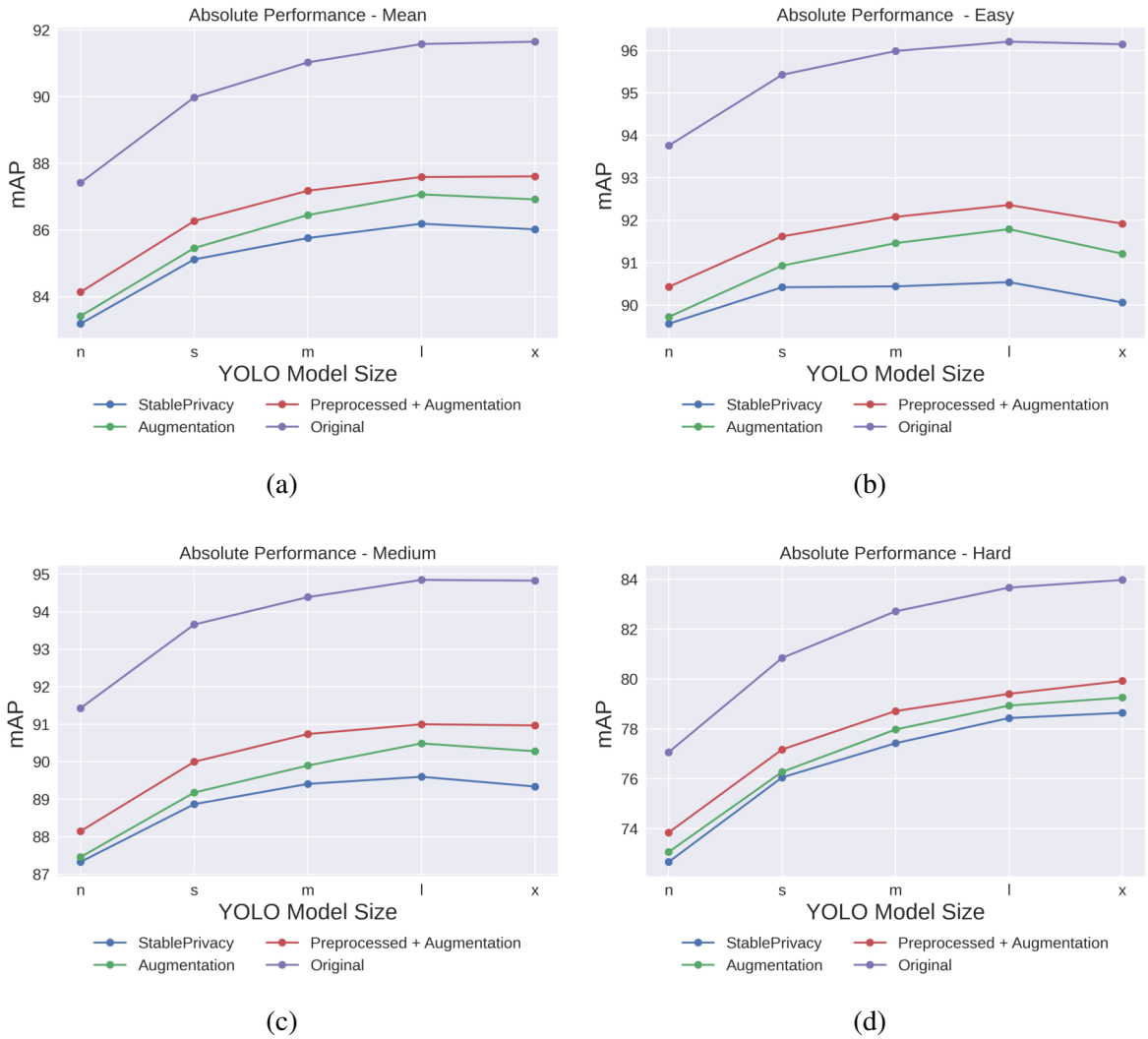


Figure 4.27: The absolute performance of YOLOv8 size variants: nano (n), small (s), medium (m), large (l) and extra-large (x), when trained on original versus data anonymized by StablePrivacy. The comparison includes three scenarios: (1) without further modifications, (2) with additional inpainting for data augmentation, and (3) with additional inpainting combined with further preprocessing of the source faces. Among these, the third scenario results in the best performance and, unlike the baseline StablePrivacy, leads to continual improvements with model size similar to when training on original data.

performance drop partly stems from the model overfitting to inpainting artifacts produced by Stable Diffusion, such as frequency-domain patterns [Corvi et al., 2023, 2022]. These artifacts may make the model predict bounding boxes based on artificial cues, ultimately hindering its ability to generalize to non-anonymized data.

To mitigate this effect, we applied Stable Diffusion to modify additional regions of the anonymized training images outside of the face area, as illustrated

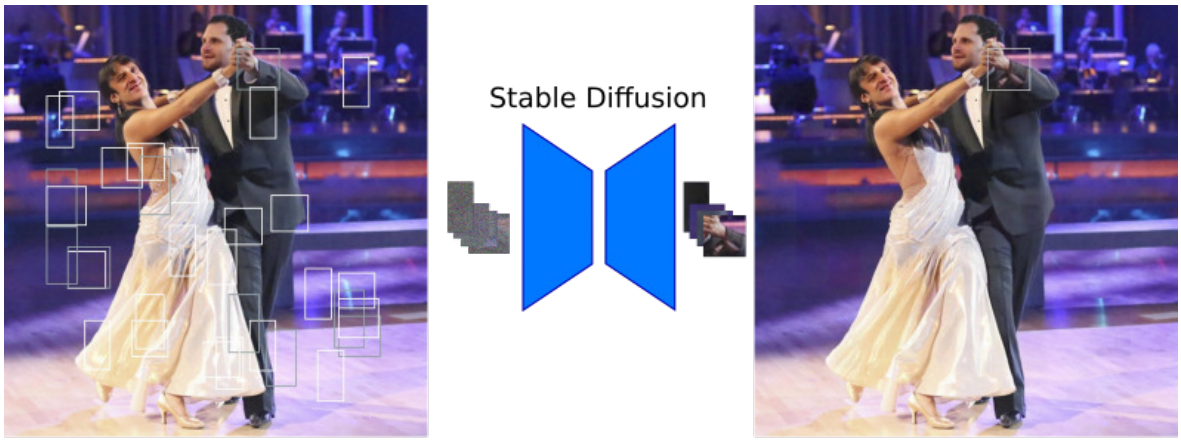


Figure 4.28: Additional inpainting is applied to the training images outside the facial regions to prevent the face detection model from learning to associate inpainting-specific artifacts with faces. The white and gray boxes in the left image illustrate the regions we process. They are cut out and sequentially passed to Stable Diffusion. Using a shortened forward diffusion process (SDEdit method) to only add limited noise to the image ensures the resulting images resemble the originals. These then replace the associated region in the larger image.

in Figure 4.28. We subsequently retrained our detection models with the goal to promote a clearer separation between inpainting artifacts and genuine facial features. For each training image, we first applied k-means clustering ($k = 3$) to the ground truth face bounding boxes to generate candidate sizes for the inpainting area. Next, a region, sized to match one of the candidates, yet with the position chosen at random while avoiding overlap with any face bounding box, was cut out and passed to Stable Diffusion for inpainting. Again, we use the SDEdit technique, shortening the forward diffusion (strength 0.7) to retain structural guidance. Thus, the generated images resemble the original, but contain the same inpainting-method-specific artifacts as the anonymized faces. We repeat this step between one and fifty times for each image.
As shown in Table 4.10 (“Augmentation”), this only leads to a slightly higher mAP for the smallest model, but leads to greater improvements for larger models. The performance of the extra-large model improved on average by 0.90 % mAP across all difficulty levels, with the gain being more pronounced for “easy” faces and less substantial for “hard” faces. We hypothesize that details within the anonymized facial region, whether realistic features such as the nose and eyes, or inpainting-specific artifacts, play a more significant role in detecting large, “easy” faces. In contrast, for smaller, more challenging

Data	Model	Difficulty			Mean ↑ (Δ)
		Easy ↑ (Δ)	Medium ↑ (Δ)	Hard ↑ (Δ)	
Augmentation	YOLOv8n	89.72 (0.16)	87.46 (0.13)	73.07 (0.40)	83.42 (0.23)
	YOLOv8s	90.93 (0.51)	89.18 (0.31)	76.28 (0.22)	85.46 (0.35)
	YOLOv8m	91.46 (1.02)	89.90 (0.49)	77.98 (0.55)	86.45 (0.69)
	YOLOv8l	91.79 (1.25)	90.49 (0.89)	78.94 (0.50)	**87.07** (0.88)
	YOLOv8x	91.21 (1.15)	90.28 (0.94)	79.26 (0.61)	86.92 (0.90)
	Average	91.02 (0.82)	89.46 (0.55)	77.11 (0.46)	85.86 (0.61)
Augmentation + Preprocessing	YOLOv8n	90.43 (0.87)	88.15 (0.82)	73.85 (1.18)	84.14 (0.96)
	YOLOv8s	91.62 (1.20)	90.00 (1.13)	77.18 (1.12)	86.27 (1.15)
	YOLOv8m	92.08 (1.64)	90.74 (1.33)	78.72 (1.29)	87.18 (1.42)
	YOLOv8l	92.36 (1.82)	91.00 (1.40)	79.41 (0.97)	87.59 (1.40)
	YOLOv8x	91.92 (1.86)	90.97 (1.63)	79.93 (1.28)	**87.61** (1.59)
	Average	91.68 (1.48)	90.17 (1.26)	77.82 (1.17)	86.56 (1.30)

Table 4.10: Data utility retention measured with the mAP of YOLOv8 face detection models of different size (n, s, m, l, x). Training data was anonymized with StablePrivacy. In the "Augmentation" variant, additional inpainting was applied, while the "Augmentation + Preprocessing" variant also preprocesses source faces. The values in brackets (Δ) describe the performance difference to the baseline StablePrivacy.

faces, contextual information outside the inpainted area, such as the person's body, becomes increasingly important. Therefore, inpainting artifacts have a reduced impact on detecting these harder cases, thus limiting the effect of our augmentation method at this level of difficulty. This interpretation is supported by the class activation maps illustrated in Figure 4.29. They show that for large faces, the region most important for the model's prediction is a small part in the center of the face, while for smaller faces, the whole face and even, to a lesser extent, the surrounding area are considered.

Notably, as demonstrated in Figure 4.26, this augmentation strategy slows the increase of the performance gap with model size.

To assess whether the observed improvements were due to mitigating the association of inpainting artifacts with faces, or simply a byproduct of general data augmentation, we applied the same method to the WIDER FACE evaluation data instead of the anonymized training set. Our experiments show that the effect of altering the validation data on the models trained on original data is negligible, with a 0.04 % decrease in performance on average across difficulty levels and model sizes. In contrast, models trained on anonymized data exhibited a 1.41 % mAP decline on average, suggesting that artifacts in the additionally inpainted regions misled the model into producing worse

Figure 4.29: The top images in each set show the face bounding box predictions made by YOLOv8 large, trained on the original data (left). The corresponding class activation map (right) uses a red-to-blue color scale, with red indicating areas of highest influence on the model's prediction. For large faces, a relatively small area is used to make the decision. The bottom images show the same results for the model trained on anonymized data. All images are from the original validation split of WIDER FACE (see Section 3.2.1), only training data is anonymized.

predictions. These findings support our hypothesis that specific artifacts introduced by the inpainting method influence face detection in models trained on anonymized data.

Moreover, we find that combining the above augmentation strategy with additionally preprocessing source faces with MTCNN [Zhang et al., 2016], cutting out the image tightly around the face and resizing it to 112 × 112 pixels significantly improves data utility by 1.30 % mAP on average (cf. Table 4.10 "Augmentation + Preprocessing"). This combined strategy not only leads to the lowest performance gap (see Figure 4.26), but also results in the mAP continually improving with model size, resembling the trend observed for the original data (cf. Figure 4.27) with the exception of the easy difficulty level.

4.3.7 Analysis of the Role of the Source Library Size

In this section, we analyze the influence the number of sources has on privacy protection and image quality. Understanding this allows us to consider alternative methods to build the source library for which only a limited number of images can be acquired. Changing the variety of source faces could influence the diversity of the anonymized images, impacting data utility, or affect privacy protection, which depends on the availability of a source face that is sufficiently dissimilar from the target. To explore this, we anonymized the WIDER FACE training dataset with StablePrivacy, varying the number of images in the source library, using either 1, 100 or 2000. The effect on utility retention can be seen in Table 4.11, while Table 4.12 presents the impact on privacy protection.

Only when using an extremely small number of source faces, namely one, utility suffers (cf. Table 4.11), with mAP decreasing on average across difficulty levels and model sizes by 0.69 %. Notably, this affects the detection of the larger, easy faces more severely than of the smaller, harder faces, further indicating that detailed features given by a variety of faces are more relevant in detecting easy faces and less in detecting hard faces. With 100 source faces, the performance is nearly equivalent to the baseline using 1000, and increasing to 2000 provides no additional improvement.

Moreover, Table 4.12 shows that changing the number of source faces only has limited influence on privacy protection. Using 2000 sources reduces the

Nr. of Sources	Difficulty			Mean ↑ (Δ)
	Easy ↑ (Δ)	Medium ↑ (Δ)	Hard ↑ (Δ)	
1 Source	89.18 (-1.03)	88.13 (-0.78)	76.39 (-0.26)	84.56 (-0.69)
100 Sources	90.34 (0.14)	88.98 (0.07)	76.82 (0.17)	85.38 (0.13)
2000 Sources	90.04 (-0.16)	88.70 (-0.21)	76.51 (-0.14)	85.08 (-0.17)

Table 4.11: Data utility retention for different numbers of StyleGAN2 source faces (1, 100, 2000). Performance is measured with the average mAP of YOLOv8 face detection models of the different sizes (n, s, m, l, x) when applied to WIDER FACE with difficulty levels easy, medium and hard. The performance difference (Δ) is relative to the model trained on data anonymized with the baseline StablePrivacy using 1000 sources. Only using one face negatively impacts data utility retention, but when using 100 or more source faces, the effect is negligible.

Model	FaceNet (↓) [%]	ArcFace (↓) [%]
StablePrivacy Baseline	0.70 ± 0.48	0.9 ± 0.33
1 Source	1.17 ± 0.58	0.90 ± 0.56
100 Sources	0.80 ± 0.27	0.67 ± 0.33
2000 Sources	0.77 ± 0.26	0.52 ± 0.10

Table 4.12: TAR when varying the number of StyleGAN2-generated source faces. Although using 2000 source faces yields a slightly stronger protection, employing fewer still offers good privacy.

TAR for ArcFace, but the trend is not clear, as performance on FaceNet almost stays the same. However, even using only 1 or 100 sources results in strong privacy protection similar to the baseline of 1000.
In conclusion, smaller source libraries of about 100 faces can be used without significantly compromising data utility or privacy protection.

4.3.8 Analysis of the Effect of Alternative Source Libraries

As the experiments discussed in the last section have shown that the number of source faces has limited influence on StablePrivacy, it seems reasonable to explore alternative source libraries even if only a few faces are available. At present, StablePrivacy leverages faces synthesized by StyleGAN2, leaving some risk of identity leakage from its training data to the final image, which can again cause privacy concerns (cf. Section 2.5.3 for a detailed discussion). To avoid this risk, images of a small group of volunteers could be used for the source library instead. As long as only a limited number of people is involved, this could be a practical alternative strategy because managing consent remains feasible. Additionally, unlike artificial images, real images do not contain artifacts that can be transferred to the anonymized images and diminish data utility. To assess the potential benefits, we replace the synthetic sources with 100 images of real people from the FFHQ dataset [Karras et al., 2021b].
Another alternative strategy to address the issue of identity leakage could be to use computer graphics-generated faces. To evaluate this, we employ 100 faces from the Face Synthetics dataset [Wood et al., 2021], which were created using techniques inspired by the visual effects industry. While these faces do not look perfectly realistic, they have been shown to bridge the domain gap between real and computer graphics-generated images well enough to

Figure 4.30: Left: examples from the two alternative source libraries used for StablePrivacy in this section. The images in the top row are real faces from FFHQ [Karras et al., 2021b], the images in the bottom row are computer graphics-generated faces from the Face Synthetics dataset [Wood et al., 2021]. Right: images from LFW (see Section 3.2.1) anonymized using the sources on the left.

be used as training data for learning to parse real faces [Wood et al., 2021]. Qualitatively, the effect on de-identification can be seen in Figure 4.30, which shows images from both alternative source libraries as well as examples of faces from the LFW dataset anonymized with them. Even though image quality seems to suffer somewhat when using sources from Face Synthetics, both strategies can create realistic outputs that look significantly different from the original. The qualitative impact of these alternatives on privacy protection can be seen in Table 4.13. It demonstrates that TAR values only vary within the margin of error from the baseline results. On the other hand, the effect on data utility is more significant (see Table 4.14). For real faces from FFHQ, mAP improved most for easy faces (1.10 %), and less for the higher difficulty levels, with 0.81 % for medium faces and 0.46 % mAP for hard ones. On average across difficulty levels and models, the utility gain was 0.79 %. In contrast to real sources, the overall effect on utility when using Face Synthetics is

Model	FaceNet (↓) [%]	ArcFace (↓) [%]
StablePrivacy Baseline	0.70 ± 0.48	0.9 ± 0.33
FFHQ Sources	0.90 ± 0.40	0.87 ± 0.37
Face Synthetics Sources	0.63 ± 0.27	0.70 ± 0.43

Table 4.13: Influence on privacy protection measured via TAR when using alternative real (FFHQ) or computer graphics-generated (Face Synthetics) faces as sources.

Alternative Sources	Easy ↑ (Δ)	Medium ↑ (Δ)	Hard ↑ (Δ)	Mean ↑ (Δ)
FFHQ	91.30 (1.10)	89.72 (0.81)	77.11 (0.46)	86.04 (0.79)
Face Synthetics	88.49 (-1.72)	87.87 (-1.04)	76.33 (-0.32)	84.23 (-1.03)

Table 4.14: Comparison of utility retention when using alternative source faces (FFHQ and Face Synthetics) to guide StablePrivacy for creating the anonymized training dataset. Performance is measured with the average across the five size variants of YOLOv8 for the different levels of difficulty of the WIDER FACE dataset. The performance difference (Δ) is relative to models trained on data anonymized with the baseline StablePrivacy.

negative, with mAP decreasing by 1.03 % on average across models. However, a similar trend of reduced impact with increasing difficulty can be observed, with performance decreasing by 1.72 % on average across model sizes for easy, 1.04 % for medium and only 0.32 % for hard faces. This is in accordance with our previous observations from Sections 4.3.6 and 4.3.7 that artifacts affect detection differently for the individual difficulties. Figures 4.31 and 4.32 show examples of the anonymized WIDER FACE training data used for this experiment.

4.3.9 Ablation Study

In this section, we systematically evaluate our approach by means of an ablation study, removing each of the following key components: IP-Adapter, source selection, and SDEdit. The results in Table 4.15 demonstrate the impact on image quality (KID and FID) and de-identification (FaceNet, ArcFace). First, omitting the IP-Adapter strongly reduces privacy protection, with the TAR climbing to 24.30 % for FaceNet and 19.89 % for ArcFace. This suggests

De-ID Method	FaceNet (↓) [%]	ArcFace (↓) [%]	KID (↓)	FID (↓)
StablePrivacy	0.70 ± 0.48	0.90 ± 0.33	0.0017 ± 0.0003	3.36
w/o IP-Adapter	24.30 ± 1.89	19.89 ± 2.92	0.0007 ± 0.0002	1.94
w/o source selection	2.2 ± 0.54	2.00 ± 1.06	0.0050 ± 0.0005	6.10
w/o SDEdit	0.60 ± 0.25	0.53 ± 0.27	0.0028 ± 0.0004	4.56

Table 4.15: Ablation Study: Evaluating the impact on privacy (FaceNet and ArcFace) and image quality (KID and FID) when removing IP-Adapter, source selection, or SDEdit from StablePrivacy. The results confirm that all three components contribute to achieving the desired balance between privacy protection and data utility.

Figure 4.31: Examples from WIDER FACE (see Section 3.2.1) anonymized with StablePrivacy using real faces from FFHQ as sources.

Figure 4.32: Examples from WIDER FACE (see Section 3.2.1) anonymized with StablePrivacy using computer graphics-generated faces from the Face Synthetic dataset.

that without the IP-Adapter, StablePrivacy tends to reconstruct the original face too closely based on the noise-modified input image. Even though this significantly improves image quality measured by KID from 0.0017 to 0.0007, the high TAR makes this choice unacceptable for de-identification.
Next, randomly choosing sources instead of using source selection according to distance in the face similarity space leads to a moderate increase in TAR to 2.2 %, highlighting the source selection process' role in enhancing privacy protection. Surprisingly, this also leads to a significantly higher KID and FID indicating lower image quality.
Finally, using random noise instead of leveraging structural guidance by adding limited noise to the original image with SDEdit before processing with Stable Diffusion slightly enhances the TAR to 0.60 % for FaceNet and 0.53 % for ArcFace. However, this comes at the expense of substantially reduced image quality, with KID rising to 0.0028.
Overall, these findings confirm that all three components, IP-Adapter, source selection and SDEdit, are essential to achieve the desired balance between privacy protection and data utility.

4.3.10 Limitations

Our previous experiments have shown that StablePrivacy can robustly create high-quality outputs (see Figure 4.18 in Section 4.3.2) and retain the features necessary for training a face detector on anonymized images (cf. Figure 4.15 in Section 4.3.2). Still, the WIDER FACE dataset contains a variety of challenging images causing unrealistic output (cf. Figure 4.33). The first image in the top row presents faces in a variety of unusual poses, typical for sports scenes, resulting in heavily distorted anonymized faces. The image in the middle (bottom) of that row shows a couple with faces in very close proximity to each other, causing the bounding boxes defining the area for inpainting to intersect. As these are processed one after the other, this causes artifacts in the overlapping region. The image above is originally blurry, while the faces anonymized by StablePrivacy are sharper than the surroundings. The rightmost image of the first row originally showed a man looking away from the camera. Due to the bias of the model for faces looking in the direction of the camera, it was inpainted in an unrealistic angle.

Figure 4.33: Example images from WIDER FACE (see Section 3.2.1) de-identified with StablePrivacy. Unusual poses, blurry images, extremely small faces and overlapping inpainting areas can cause low-quality anonymizations.

An additional limitation of our approach is the runtime: It takes 3.42 seconds to de-identify a single face on average on our hardware (Nvidia Quadro GV100) when a strength of 0.7 is used. While this runtime is comparable to LDFA [Klemp et al., 2023], the other diffusion-based approach, it falls far short of the speed of GAN-based methods and can be prohibitively slow for large datasets. For the 13,233 faces in LFW this adds up to 12.57 hours and to 151.42 hours for WIDER FACE (159,393 faces). However, StablePrivacy can be easily processed on multiple GPUs in parallel as the anonymization process is completely independent for each image. Moreover, we discussed strategies to speed up anonymization by adjusting strength according to face size and privacy requirements in Section 4.3.5. Additionally, recent work, such as the development of Latent Consistency Models (see Section 2.5.2) promises to reduce the computational cost of using Stable Diffusion by substantially reducing the number of backward diffusion steps necessary to create high-quality images. Yet, our experiments using the Hugging Face implementation

of LCM reveal that although runtime is significantly decreased, image quality suffers with the FID increasing from 3.36 without LCM to 8.73 or 8.75, depending on the number of diffusion steps (see Table 4.16). Moreover, as shown in Figure 4.34, many of the generated images exhibit obvious artifacts. Finally, we want to point out that while we took care to balance the source library with respect to gender, there are other biases such as skin color or age that are not considered in the current work.

De-ID Method	FaceNet (↓) [%]	ArcFace (↓) [%]	KID (↓)	FID (↓)
StablePrivacy	0.70 ± 0.48	0.90 ± 0.33	0.0017 ± 0.0003	3.36
LCM 10 steps	0.57 ± 0.26	0.60 ± 0.38	0.0067 ± 0.0006	8.73
LCM 6 steps	0.30 ± 0.31	0.57 ± 0.26	0.0073 ± 0.0006	8.75

Table 4.16: Image quality is significantly reduced when LCM is used to speed up the anonymization process. Privacy protection improves, but probably due to unrealistic deformation of faces.

Figure 4.34: Images from the LFW dataset (see Section 3.2.1) anonymized with StablePrivacy when using the LCM technique to reduce the number of diffusion steps to ten (top row) or six (bottom row).

Chapter 5

Summary and Outlook

In this thesis we propose two novel approaches to synthesis-based face de-identification: DetailedPrivacy and StablePrivacy. Each achieves a unique balance in the privacy – data utility trade-off according to the requirements of their use case. Furthermore, they differ in the data they can be applied to.
DetailedPrivacy (cf. Section 4.2) can retain the precise face pose and expression from the original to its surrogate. Compared to state-of-the-art approaches, it stands out for its strong preservation of expression, while the pose retention is limited by weak results for the pitch angle. In terms of privacy protection, it trails behind other approaches like CIAGAN and StablePrivacy but achieves comparable performance to DeepPrivacy, DeepPrivacy2 and LDFA (see Section 4.3.1, Table 4.5). An additional constraint of this approach is its limited usefulness for smaller faces for which the extraction of detailed landmarks is not possible. A key advantage, on the other hand, is that it can be applied to video data, as it creates temporally coherent faces for consecutive frames.
In contrast to DetailedPrivacy, StablePrivacy focuses on transferring human appearance only to the anonymized images, resulting in exceptionally strong privacy protection that surpasses all other tested approaches. Moreover, it produces high-quality results even for small, occluded or otherwise challenging faces. This makes it ideal for anonymizing complex face detection datasets meant for training deep learning-based models, as demonstrated in Section 4.3.2.
Both StablePrivacy and DetailedPrivacy are limited to de-identifying faces. Therefore, it is important to note that humans can be identified by features outside the face, for example, by their gait [Cutting and Kozlowski, 1977;

Stevenage et al., 1999; dos Santos et al., 2022], unique tattoos or scars. To address this issue, CIAGAN [Maximov et al., 2020] and DeepPrivacy2 [Hukkelås and Lindseth, 2023] also explore full-body anonymization, aside from face de-identification. StablePrivacy could in the future also be extended in this way, possibly without retraining, as its generative components, IP-Adapter and Stable Diffusion, are already trained on diverse datasets featuring entire human bodies. Should fine-tuning or retraining of the IP-Adapter module prove necessary, the Flicker Diverse Humans (FDH) dataset provided by Hukkelås and Lindseth [2023] could be a suitable resource. However, as full-body anonymization changes a larger area of the original image, it will probably have a negative influence on data utility. This effect can be observed in our experiments with an increased inpainting area in Section 4.3.4 (see Figure 4.24).
An additional privacy concern for both de-identification approaches is the potential identity leakage from the dataset the generative method was trained on to the final anonymized image, as discussed in Section 2.5.3. To promote the real-world usage of our approaches, it would be beneficial to quantify this risk. For avoiding this issue, we suggested using alternative source libraries in Section 4.3.8. Future work could investigate the usage of additional computer graphics-generated source libraries to determine whether data utility can be improved.
Another possible direction of research could be to further explore the influence of landmarks on the privacy – data utility trade-off. The FSGANv2 component of DetailedPrivacy could be retrained to utilize different numbers of landmarks to observe the effect on the metrics discussed in Section 4.2. Similarly, StablePrivacy could be extended, for instance, by using a customized ControlNet [Zhang et al., 2023] to include guidance from a varying number of landmarks. Furthermore, it would be interesting to evaluate whether StablePrivacy is suitable for de-identifying data for other computer vision tasks, such as human keypoints detection or action recognition. Adding guidance from a limited number of keypoints to StablePrivacy (i.e., with ControlNet) could further increase the utility for these applications. Yet, the potential influence on privacy must be considered.

Overall, to foster the widespread usage of anonymization for a broad range of

datasets in computer vision, it is crucial to demonstrate high levels of utility. To this end, de-identification approaches need to be further refined to close the performance gap between original and anonymized data.

Bibliography

Martin Abadi, Andy Chu, Ian J Goodfellow, H B McMahan, Ilya Mironov, Kunal Talwar, and Li Zhang. Deep learning with differential privacy. *Conference on Computer and Communications Security*, 2016.

Naphtali Abudarham and Galit Yovel. Reverse engineering the face space: Discovering the critical features for face identification. *Journal of Vision*, 16 3:40, 2014.

Timo Ahonen, Abdenour Hadid, and Matti Pietikäinen. Face description with local binary patterns: Application to face recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28:2037–2041, 2006.

Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein generative adversarial networks. *International Conference on Machine Learning*, 70:214–223, 2017.

Australian Government. Privacy act 1988. URL `https://www.legislation.gov.au/C2004A03712/latest/text`. Accessed: 2025-04-09.

Jimmy Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. Layer normalization. *ArXiv*, 2016.

Y. Bengio, P. Simard, and P. Frasconi. Learning long-term dependencies with gradient descent is difficult. *IEEE Transactions on Neural Networks*, 5:157–166, 1994.

Tamara L. Berg, Alexander C. Berg, Jaety Edwards, and David Alexander Forsyth. Who's in the picture. *Conference on Neural Information Processing Systems*, 2004.

Eyal Betzalel, Coby Penso, Aviv Navon, and Ethan Fetaya. A study on the evaluation of generative models. *ArXiv*, 2022.

Abeba Birhane and Vinay Uday Prabhu. Large image datasets: A pyrrhic win for computer vision? *Winter Conference on Applications of Computer Vision*, pages 1536–1546, 2021.

Dmitri Bitouk, Neeraj Kumar, Samreen Dhillon, Peter N. Belhumeur, and Shree K. Nayar. Face swapping: automatically replacing faces in photographs. *ACM SIGGRAPH*, 2008.

Mikołaj Bińkowski, Danica J Sutherland, Michael Arbel, and Arthur Gretton. Demystifying MMD GANs. *International Conference on Learning Representations*, 2018.

Fadi Boutros, Marco Huber, Patrick Siebke, Tim Rieber, and Naser Damer. SFace: Privacy-friendly and accurate face recognition using synthetic data. *International Joint Conference on Biometrics*, pages 1–11, 2022.

Fadi Boutros, Vitomir Struc, Julian Fierrez, and Naser Damer. Synthetic data for face recognition: Current state and future prospects. *Image and Vision Computing*, 135:104688, 2023.

Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale GAN training for high fidelity natural image synthesis. *International Conference on Learning Representations*, 2019.

Holger Caesar, Varun Bankiti, Alex H. Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yuxin Pan, Giancarlo Baldan, and Oscar Beijbom. nuScenes: A multimodal dataset for autonomous driving. *Conference on Computer Vision and Pattern Recognition*, 2020.

California Legislative Counsel. Assembly Bill No. 375 – Chapter 55: California Consumer Privacy Act of 2018, 2018. URL `https://leginfo.legislature.ca.gov/faces/billTextClient.xhtml?bill_id=201720180AB375`. Accessed: 2024-03-19.

Zhimin Cao, Qi Yin, Xiaoou Tang, and Jian Sun. Face recognition with learning-based descriptor. *Conference on Computer Vision and Pattern Recognition*, pages 2707–2714, 2010.

Nicolas Carlini, Jamie Hayes, Milad Nasr, Matthew Jagielski, Vikash Sehwag, Florian Tramer, Borja Balle, Daphne Ippolito, and Eric Wallace. Extracting training data from diffusion models. In *USENIX Security Symposium*, pages 5253–5270, 2023.

Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin P Murphy, and Alan Loddon Yuille. DeepLab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40:834–848, 2016.

Renwang Chen, Xuanhong Chen, Bingbing Ni, and Yanhao Ge. SimSwap: An efficient framework for high fidelity face swapping. *ACM International Conference on Multimedia*, 2020a.

Tianshui Chen, Tao Pu, Yuan Xie, Hefeng Wu, Lingbo Liu, and Liang Lin. Cross-domain facial expression recognition: A unified evaluation benchmark and adversarial graph learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020b.

Saheb Chhabra, Richa Singh, Mayank Vatsa, and Gaurav Gupta. Anonymizing k-facial attributes via adversarial perturbations. *International Joint Conference on Artificial Intelligence*, 2018.

Durkhyun Cho, Jin Han Lee, and Il Hong Suh. CLEANIR: Controllable attribute-preserving natural identity remover. *Applied Sciences*, 2020.

Zhiguang Chu, Jingsha He, Dongdong Peng, Xing Zhang, and Nafei Zhu. Differentially private denoise diffusion probability models. *IEEE Access*, 11:108033–108040, 2023.

Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine learning*, 20:273–297, 1995.

Riccardo Corvi, Davide Cozzolino, Giada Zingarini, Giovanni Poggi, Koki Nagano, and Luisa Verdoliva. On the detection of synthetic images generated by diffusion models. *International Conference on Acoustics, Speech and Signal Processing*, pages 1–5, 2022.

Riccardo Corvi, Davide Cozzolino, Giovanni Poggi, Koki Nagano, and Luisa Verdoliva. Intriguing properties of synthetic images: from generative adversarial networks to diffusion models. *Conference on Computer Vision and Pattern Recognition Workshops*, pages 973–982, 2023.

James E. Cutting and Lynn T. Kozlowski. Recognizing friends by their walk: Gait perception without familiarity cues. *Bulletin of the psychonomic society*, 9:353–356, 1977.

Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *Conference on Computer Vision and Pattern Recognition*, volume 1, pages 886–893, 2005.

Graham Davies, Hadyn D Ellis, and John W Shepherd. Cue saliency in faces as assessed by the 'photofit' technique. *Perception*, 6:263 – 269, 1977.

Anis Davoudi, Kumar Rohit Malhotra, Benjamin Shickel, Scott Siegel, Seth Williams, Matthew M Ruppert, Emel Bihorac, Tezcan Ozrazgat-Baslanti, Patrick James Tighe, Azra Bihorac, and Parisa Rashidi. Intelligent ICU for autonomous patient monitoring using pervasive sensing and deep learning. *Scientific Reports*, 9, 2019.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. *Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009.

Jiankang Deng, Jia Guo, Niannan Xue, and Stefanos Zafeiriou. ArcFace: Additive angular margin loss for deep face recognition. *Conference on Computer Vision and Pattern Recognition*, pages 4690–4699, 2019.

Jiankang Deng, Jia Guo, Evangelos Ververas, Irene Kotsia, and Stefanos Zafeiriou. RetinaFace: Single-shot multi-level face localisation in the wild. In *Conference on Computer Vision and Pattern Recognition*, pages 5202–5211, 2020.

Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Conference on Neural Information Processing Systems*, 34:8780–8794, 2021.

Claudio Filipi Gonçalves dos Santos, Diego de Souza Oliveira, Leandro A. Passos, Rafael Gonçalves Pires, Daniel Felipe Silva Santos, Lucas Pascotti Valem, Thierry P. Moreira, Marcos Cleison S. Santana, Mateus Roder, Jo Paulo Papa, and Danilo Colombo. Gait recognition based on deep learning: A survey. *ACM Computing Surveys*, 55:1 – 34, 2022.

D. C. Dowson and B. V. Landau. The Fréchet distance between multivariate normal distributions. *Journal of Multivariate Analysis*, 12:450–455, 1982.

Liang Du, Meng Yi, Erik Blasch, and Haibin Ling. GARP-face: Balancing privacy protection and utility preservation in face de-identification. *IEEE International Joint Conference on Biometrics*, pages 1–8, 2014.

John C. Duchi, Elad Hazan, and Yoram Singer. Adaptive subgradient methods for online learning and stochastic optimization. In *Conference on Learning Theory*, pages 257–269, 2010.

Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam D. Smith. Calibrating noise to sensitivity in private data analysis. *Journal of Privacy and Confidentiality*, 7:17–51, 2006.

Patrick Esser, Robin Rombach, and Björn Ommer. Taming transformers for high-resolution image synthesis. In *Conference on Computer Vision and Pattern Recognition*, pages 12868–12878, 2020.

European Union. Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data (General Data Protection Regulation), 2016. URL `https://eur-lex.europa.eu/eli/reg/2016/679/oj`. Accessed: 2025-04-09.

Mark Everingham, Andrew Zisserman, Christopher K. I. Williams, Luc Van Gool, Moray Allan, Christopher M. Bishop, Olivier Chapelle, Navneet Dalal, Thomas Deselaers, Gyuri Dorkó, et al. The 2005 pascal visual object classes challenge. *Machine Learning Challenges, Evaluating Predictive Uncertainty, Visual Object Classification and Recognizing Textual Entailment, First PASCAL Machine Learning Challenges Workshop*, pages 117–176, 2005.

Mark Everingham, L. Van Gool, Christopher K. I. Williams, John Winn, and Andrew Zisserman. The pascal visual object classes challenge 2007 (voc 2007) results (2007), 2008.

Gabriele Fanelli, Matthias Dantone, Juergen Gall, Andrea Fossati, and Luc Van Gool. Random forests for real time 3D face analysis. *International Journal of Computer Vision*, 2013.

Vitaly Feldman and Chiyuan Zhang. What neural networks memorize and why: Discovering the long tail via influence estimation. *Conference on Neural Information Processing Systems*, 33:2881–2891, 2020.

Pedro F. Felzenszwalb, Ross B. Girshick, David McAllester, and Deva Ramanan. Object detection with discriminatively trained part-based models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32:1627–1645, 2010.

Qianli Feng, Chen Guo, Fabian Benitez-Quiroz, and Aleix M Martínez. When do GANs replicate? on the choice of dataset size. *International Conference on Computer Vision*, pages 6681–6690, 2021.

Oran Gafni, Lior Wolf, and Yaniv Taigman. Live face de-identification in video. *International Conference on Computer Vision*, 2019.

Zheng Ge, Songtao Liu, Feng Wang, Zeming Li, and Jian Sun. YOLOX: Exceeding YOLO series in 2021. *ArXiv*, abs/2107.08430, 2021.

Robert Geirhos, Patricia Rubisch, Claudio Michaelis, Matthias Bethge, Felix A. Wichmann, and Wieland Brendel. ImageNet-trained CNNs are biased towards texture; increasing shape bias improves accuracy and robustness. In *International Conference on Learning Representations*, 2019.

Jacob Gildenblat. Pytorch library for CAM methods, 2021.

Ross B. Girshick. Fast R-CNN. In *International Conference on Computer Vision*, 2015.

Ross B. Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. *Conference on Computer Vision and Pattern Recognition*, pages 580–587, 2013.

Xavier Glorot and Yoshua Bengio. Understanding the difficulty of training deep feedforward neural networks. In *International Conference on Artificial Intelligence and Statistics*, volume 9, pages 249–256, 4 2010.

Ian J Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *Conference on Neural Information Processing Systems*, 2014.

Ralph Gross, Edoardo M. Airoldi, Bradley A. Malin, and Latanya Sweeney. Integrating utility into face de-identification. In *International Symposium on Privacy Enhancing Technologies*, 2005.

Ralph Gross, Latanya Sweeney, Fernando De la Torre, and Simon Baker. Model-based face de-identification. *Conference on Computer Vision and Pattern Recognition Workshop*, page 161, 2006.

Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron C. Courville. Improved training of Wasserstein GANs. In *Conference on Neural Information Processing Systems*, pages 5767–5777, 2017.

Jia Guo, Jiankang Deng, Alexandros Lattas, and Stefanos Zafeiriou. Sample and computation redistribution for efficient face detection. In *International Conference on Learning Representations*, 2022.

Abdenour Hadid. The local binary pattern approach and its applications to face analysis. *Workshops on Image Processing Theory, Tools and Applications*, pages 1–9, 2008.

Kaiming He and Jian Sun. Convolutional neural networks at constrained time cost. In *Conference on Computer Vision and Pattern Recognition*, pages 5353–5360, 2014.

Kaiming He, X. Zhang, Shaoqing Ren, and Jian Sun. Spatial pyramid pooling in deep convolutional networks for visual recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37:1904–1916, 2014.

Kaiming He, X. Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016.

Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross B. Girshick. Mask R-CNN. *International Conference on Computer Vision*, pages 2980–2988, 2017.

Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Conference on Neural Information Processing Systems*, 2017.

Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *Conference on Neural Information Processing Systems Workshop*, 2021.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Conference on Neural Information Processing Systems*, 33:6840–6851, 2020.

Gary B Huang, Marwan Mattar, Tamara Berg, and Eric Learned-Miller. Labeled faces in the wild: A database for studying face recognition in unconstrained environments. *Workshop on Faces in'Real-Life'Images: Detection, Alignment, and Recognition*, 2008.

Xun Huang and Serge J Belongie. Arbitrary style transfer in real-time with adaptive instance normalization. In *International Conference on Computer Vision*, pages 1510–1519, 2017.

Y. Huang, Yuhan Wang, Ying Tai, Xiaoming Liu, Pengcheng Shen, Shaoxin Li, Jilin Li, and Feiyue Huang. CurricularFace: Adaptive curriculum learning loss for deep face recognition. *Conference on Computer Vision and Pattern Recognition*, 2020.

Hakon Hukkelas and Frank Lindseth. Does image anonymization impact computer vision training? *Conference on Computer Vision and Pattern Recognition Workshops*, pages 140–150, 2023.

Håkon Hukkelås and Frank Lindseth. DeepPrivacy2: Towards realistic full-body anonymization. *Winter Conference on Applications of Computer Vision*, pages 1329–1338, 2023.

Håkon Hukkelås, Rudolf Mester, and Frank Lindseth. DeepPrivacy: A generative adversarial network for face anonymization. *Advances in Visual Computing*, 2019.

Håkon Hukkelås, Frank Lindseth, and Rudolf Mester. Image inpainting with learnable feature imputation. In *German Conference on Pattern Recognition*, volume 12544, pages 388–403, 2020.

Håkon Hukkelås, Morten Smebye, Rudolf Mester, and Frank Lindseth. Realistic full-body anonymization with surface-guided GANs. *Winter Conference on Applications of Computer Vision*, pages 1430–1440, 2022.

Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International Conference on Machine Learning*, pages 448–456, 2015.

Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. *Conference on Computer Vision and Pattern Recognition*, pages 5967–5976, 2017.

Aleksei Grigorevich Ivakhnenko and Valentin Grigorevich Lapa. Cybernetic predicting devices. 1966.

Sadeep Jayasumana, Srikumar Ramalingam, Andreas Veit, Daniel Glasner, Ayan Chakrabarti, and Sanjiv Kumar. Rethinking FID: Towards a better evaluation metric for image generation. *Conference on Computer Vision and Pattern Recognition*, pages 9307–9315, 2023.

Glenn Jocher. Ultralytics YOLOv5, 2020. URL `https://github.com/ultralytics/yolov5`. Accessed: 2025-04-09.

Glenn Jocher, Ayush Chaurasia, and Jing Qiu. Ultralytics YOLOv8, 2023. URL `https://github.com/ultralytics/ultralytics`. Accessed: 2025-04-09.

Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *European Conference Computer Vision*, volume 9906, pages 694–711, 2016.

Indu Joshi, Marcel Grimmer, Christian Rathgeb, Christoph Busch, François Brémond, and Antitza Dantcheva. Synthetic data in human analysis: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46:4957–4976, 2022.

Won Kyung Jung and Hun Yeong Kwon. Privacy and data protection regulations for ai using publicly available data: Clearview AI case. In *International Conference on Theory and Practice of Electronic Governance*, pages 48–55, 2024.

Animesh Karnewar and Oliver Wang. MSG-GAN: Multi-scale gradients for generative adversarial networks. *Conference on Computer Vision and Pattern Recognition*, pages 7796–7805, 2020.

Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of GANs for improved quality, stability, and variation. *International Conference on Learning Representations*, 2018.

Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of StyleGAN. *Conference on Computer Vision and Pattern Recognition*, 2020.

Tero Karras, Miika Aittala, Samuli Laine, Erik Härkönen, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Alias-free generative adversarial networks. *Conference on Neural Information Processing Systems*, pages 852–863, 2021a.

Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021b.

Davis E King. Dlib-ml: A machine learning toolkit. *The Journal of Machine Learning Research*, 2009.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations*, 2015.

Leonard Klein. Training machine-learning models on de-identified datasets, 2023. Bachelor Thesis.

Marvin Klemp, Kevin Rösch, Royden Wagner, Jannik Quehl, and Martin Lauer. LDFA: Latent diffusion face anonymization for self-driving applications. *Conference on Computer Vision and Pattern Recognition Workshops*, pages 3199–3205, 2023.

Sander R. Klomp, Matthew Van Rijn, Rob G. J. Wijnhoven, Cees G. M. Snoek, and Peter H. N. De With. Safe fakes: Evaluating face anonymizers for face detectors. *International Conference on Automatic Face and Gesture Recognition*, 2021.

Martin Knoche, Stefan Hörmann, and Gerhard Rigoll. Susceptibility to image resolution in face recognition and training strategies to enhance robustness. *Leibniz Transactions on Embedded Systems*, 8:01:1–01:20, 2022.

Jan Niklas Kolf, Tim Rieber, Jurek Elliesen, Fadi Boutros, Arjan Kuijper, and Naser Damer. Identity-driven three-player generative adversarial network for synthetic-based face recognition. *Conference on Computer Vision and Pattern Recognition Workshops*, pages 806–816, 2023.

Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. ImageNet classification with deep convolutional neural networks. In *Conference on Neural Information Processing Systems*, volume 25, 2012.

Solomon Kullback and R. A. Leibler. On information and sufficiency. *Annals of Mathematical Statistics*, 22:79–86, 1951.

Alina Kuznetsova, Hassan Rom, Neil Alldrin, Jasper Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Malloci, Alexander Kolesnikov, Tom Duerig, and Vittorio Ferrari. The open images dataset v4: Unified image classification, object detection, and visual relationship detection at scale. *International Journal of Computer Vision*, pages 1956 – 1981, 2020.

Yann LeCun, Bernhard E. Boser, John S. Denker, Donnie Henderson, Richard E. Howard, Wayne E. Hubbard, and Lawrence D. Jackel. Handwritten digit recognition with a back-propagation network. In *Conference on Neural Information Processing Systems*, 1989a.

Yann LeCun, Bernhard E Boser, John S Denker, Donnie Henderson, Richard E. Howard, Wayne E. Hubbard, and Lawrence D. Jackel. Backpropagation applied to handwritten zip code recognition. *Neural Computation*, 1:541–551, 1989b.

Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86:2278–2324, 1998.

Christian Ledig, Lucas Theis, Ferenc Huszar, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew P Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, and Wenzhe Shi. Photo-realistic single image super-resolution using a generative adversarial network. In *Conference on Computer Vision and Pattern Recognition*, pages 105–114, 2017.

Jun Ha Lee and Sujeong You. Balancing privacy and accuracy: Exploring the impact of data anonymization on deep learning models in computer vision. *IEEE Access*, 12:8346–8358, 2024.

Young-Shin Lee and Won-Hyung Park. Diagnosis of depressive disorder model on facial expression based on Fast R-CNN. *Diagnostics*, 12, 2022.

Andreas Leibl and Helmut Mayer. StablePrivacy: Diffusion-based privacy enhancement for face image datasets. In *British Machine Vision Conference Workshop*, 2024.

Andreas Leibl, Andreas Attenberger, Andreas Meißner, Stefan Altmann, and Helmut Mayer. De-identifying face image datasets while retaining facial expressions. *International Joint Conference on Biometrics*, 2023.

Jian Li, Yabiao Wang, Changan Wang, Ying Tai, Jianjun Qian, Jian Yang, Chengjie Wang, Jilin Li, and Feiyue Huang. DSFD: Dual shot face detector. *Conference on Computer Vision and Pattern Recognition*, pages 5055–5064, 2019.

Puheng Li, Zhong Li, Huishuai Zhang, and Jiang Bian. On the generalization properties of diffusion models. *Conference on Neural Information Processing Systems*, 36:2097–2127, 2023.

Xiang Li, Wenhai Wang, Lijun Wu, Shuo Chen, Xiaolin Hu, Jun Li, Jinhui Tang, and Jian Yang. Generalized focal loss: Learning qualified and distributed bounding boxes for dense object detection. In *Conference on Neural Information Processing Systems*, 2020.

Min Lin, Qiang Chen, and Shuicheng Yan. Network in network. *International Conference on Learning Representations*, 2013.

Tsung-Yi Lin, Michael Maire, Serge J. Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft COCO: Common objects in context. In *European Conference on Computer Vision*, 2014.

Seppo Linnainmaa. The representation of the cumulative rounding error of an algorithm as a taylor expansion of the local rounding errors, 1970. Master's Thesis.

Chengjun Liu and Harry Wechsler. Gabor feature based classification using the enhanced fisher linear discriminant model for face recognition. *IEEE Transactions on Image Processing*, 11 4:467–76, 2002.

Dongdong Liu, Bowen Liu, Tao Lin, Guangya Liu, Guoyu Yang, Dezhen Qi, Ye Qiu, Yuer Lu, Qinmei Yuan, Stella C Shuai, Xia Li, Ou Liu, Xiangdong Tang, Jianwei Shuai, Yuping Cao, and Hai Lin. Measuring depression severity based on facial expression and body movement using deep convolutional neural network. *Frontiers in Psychiatry*, 13, 2022.

Kunlin Liu, Ivan Perov, Daiheng Gao, Nikolay Chervoniy, Wenbo Zhou, and Weiming Zhang. DeepFaceLab: Integrated, flexible and extensible face-swapping framework. *Pattern Recognition*, 141: 109628, 2023.

Shu Liu, Lu Qi, Haifang Qin, Jianping Shi, and Jiaya Jia. Path aggregation network for instance segmentation. *Conference on Computer Vision and Pattern Recognition*, pages 8759–8768, 2018.

W. Liu, Dragomir Anguelov, D. Erhan, Christian Szegedy, Scott E Reed, Cheng-Yang Fu, and Alexander C Berg. SSD: Single shot multibox detector. In *European Conference on Computer Vision*, 2015a.

Weiyang Liu, Yandong Wen, Zhiding Yu, and Meng Yang. Large-margin softmax loss for convolutional neural networks. In *International Conference on Machine Learning*, 2016.

Weiyang Liu, Yandong Wen, Zhiding Yu, Ming Li, Bhiksha Raj, and Le Song. SphereFace: Deep hypersphere embedding for face recognition. *Conference on Computer Vision and Pattern Recognition*, pages 6738–6746, 2017.

Yang Liu and Xu Tang. Bfbox: Searching face-appropriate backbone and feature pyramid network for face detector. *Conference on Computer Vision and Pattern Recognition*, pages 13565–13574, 2020.

Yang Liu, Xu Tang, Junyu Han, Jingtuo Liu, Dinger Rui, and Xiang Wu. HAMBox: Delving into mining high-quality anchors on face detection. *Conference on Computer Vision and Pattern Recognition*, pages 13043–13051, 2020.

Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. *International Conference on Computer Vision*, pages 3730–3738, 2015b.

David G. Lowe. Object recognition from local scale-invariant features. In *International Conference on Computer Vision*, volume 2, pages 1150–1157, 1999.

Mario Lucic, Karol Kurach, Marcin Michalski, Sylvain Gelly, and Olivier Bousquet. Are GANs created equal? a large-scale study. In *Conference on Neural Information Processing Systems*, pages 698–707, 2018.

Simian Luo, Yiqin Tan, Longbo Huang, Jian Li, and Hang Zhao. Latent consistency models: Synthesizing high-resolution images with few-step inference. *ArXiv*, 2023a.

Simian Luo, Yiqin Tan, Suraj Patil, Daniel Gu, Patrick von Platen, Apolin'ario Passos, Longbo Huang, Jian Li, and Hang Zhao. LCM-LoRA: A universal stable-diffusion acceleration module. *ArXiv*, 2023b.

Michael Lyons, Miyuki Kamachi, and Jiro Gyoba. The japanese female facial expression (JAFFE) dataset, 1998.

Zhaoyang Lyu, Zhifeng Kong, Xudong Xu, Liang Pan, and Dahua Lin. A conditional point diffusion-refinement paradigm for 3D point cloud completion. In *International Conference on Learning Representations*, 2022.

Xudong Mao, Qing Li, Haoran Xie, Raymond Y K Lau, Zhen Wang, and Stephen Paul Smolley. Least squares generative adversarial networks. In *International Conference on Computer Vision*, pages 2813–2821, 2017.

Maxim Maximov, Ismail Elezi, and Laura Leal-Taixe. CIAGAN: Conditional identity anonymization generative adversarial networks. *Computer Society Conference on Computer Vision and Pattern Recognition*, pages 5446–5455, 2020.

Blaz Meden, Peter Rot, Philipp Terhorst, Naser Damer, Arjan Kuijper, Walter J. Scheirer, Arun Ross, Peter Peer, and Vitomir Struc. Privacy-enhancing face biometrics: A comprehensive survey. *IEEE Transactions on Information Forensics and Security*, 16:4147–4183, 2021.

Andreas Meißner, Andreas Fröhlich, and Michaela Geierhos. Keep it simple: Local search-based latent space editing. *International Joint Conference on Computational Intelligence*, pages 273–283, 2022.

Chenlin Meng, Yutong He, Yang Song, Jiaming Song, Jiajun Wu, Jun-Yan Zhu, and Stefano Ermon. SDEdit: Guided image synthesis and editing with stochastic differential equations. *International Conference on Learning Representations*, 2021.

Lily Meng, Zongji Sun, and Odette Tejada Collado. Efficient approach to de-identifying faces in videos. *IET Signal Processing*, 11:1039–1045, 2017.

Mehdi Mirza and Simon Osindero. Conditional generative adversarial nets. *ArXiv*, 2014.

Ali Mollahosseini, Behzad Hasani, and Mohammad H Mahoor. AffectNet: A database for facial expression, valence, and arousal computing in the wild. *IEEE Transactions on Affective Computing*, 10:18–31, 2019.

Saleh Mosaddegh, Loïc Simon, and Frédéric Jurie. Photorealistic face de-identification by aggregating donors' face components. In *Asian Conference on Computer Vision*, 2014.

Chong Mou, Xintao Wang, Liangbin Xie, Jing Zhang, Zhongang Qi, Ying Shan, and Xiaohu Qie. T2I-Adapter: Learning adapters to dig out more controllable ability for text-to-image diffusion models. *AAAI Conference on Artificial Intelligence*, 2023.

Mohammed Bany Muhammad and Mohammed Yeasin. Eigen-CAM: Class activation map using principal components. *International Joint Conference on Neural Networks*, pages 1–7, 2020.

Vaishnavh Nagarajan, Colin Raffel, and Ian J Goodfellow. Theoretical insights into memorization in GANs. In *Neural Information Processing Systems Workshop*, volume 1, page 3, 2018.

Vinod Nair and Geoffrey E Hinton. Rectified linear units improve restricted boltzmann machines. In *International Conference on Machine Learning*, 2010.

Milind R Naphade, John R Smith, Jelena Teić, Shih-Fu Chang, Winston H. Hsu, Lyndon S. Kennedy, Alexander Hauptmann, and Jon Curtis. Large-scale concept ontology for multimedia. *IEEE MultiMedia*, 13:86–91, 2006.

Ryota Natsume, Tatsuya Yatagawa, and Shigeo Morishima. FSNet: An identity-aware generative model for image-based face swapping. In *Asian Conference on Computer Vision*, volume 11366, pages 117–132, 2018a.

Ryota Natsume, Tatsuya Yatagawa, and Shigeo Morishima. RSGAN: Face swapping and editing using face and hair representation in latent spaces. *ACM SIGGRAPH 2018 Posters*, 2018b.

Elaine M Newton, Latanya Sweeney, and Bradley A Malin. Preserving privacy by de-identifying face images. *IEEE Transactions on Knowledge and Data Engineering*, 17:232–243, 2005.

Alex Nichol, Prafulla Dhariwal, Aditya Ramesh, Pranav Shyam, Pamela Mishkin, Bob McGrew, Ilya Sutskever, and Mark Chen. GLIDE: Towards photorealistic image generation and editing with text-guided diffusion models. In *International Conference on Machine Learning*, 2021.

Alexander Quinn Nichol and Prafulla Dhariwal. Improved denoising diffusion probabilistic models. *International Conference on Machine Learning*, 139:8162–8171, 2 2021.

Koichiro Niinuma, Itir Onal Ertugrul, Jeffrey F Cohn, and László A Jeni. Synthetic expressions are better than real for learning to detect facial actions. *Winter Conference on Applications of Computer Vision*, pages 1247–1256, 2020.

Yuval Nirkin, Yosi Keller, and Tal Hassner. FSGAN: Subject agnostic face swapping and reenactment. *International Conference on Computer Vision*, pages 7183–7192, 2019.

Yuval Nirkin, Tal Hassner, and Yosi Keller. FSGANv2: Better subject agnostic face swapping and reenactment. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, page 1, 2022.

Office of the Australian Information Commissioner. Australian privacy principles. URL `https://www.oaic.gov.au/privacy/australian-privacy-principles`. Accessed: 2025-04-09.

Hatef Otroshi-Shahreza and Sébastien Marcel. Unveiling synthetic faces: How synthetic datasets can expose real identities. *ArXiv*, 2024.

Zhaoqing Pan, Weijie Yu, Bosi Wang, Haoran Xie, Victor S Sheng, Jianjun Lei, and Sam Kwong. Loss functions of generative adversarial networks (GANs): Opportunities and challenges. *IEEE Transactions on Emerging Topics in Computational Intelligence*, 4:500–522, 2020.

Deepak Pathak, Philipp Krähenbühl, Jeff Donahue, Trevor Darrell, and Alexei A Efros. Context encoders: Feature learning by inpainting. In *Conference on Computer Vision and Pattern Recognition*, pages 2536–2544, 2016.

Amandalynne Paullada, Inioluwa Deborah Raji, Emily M. Bender, Emily L. Denton, and A. Hanna. Data and its (dis)contents: A survey of dataset development and use in machine learning research. *Patterns*, 2, 2020.

William Peebles and Saining Xie. Scalable diffusion models with transformers. In *International Conference on Computer Vision*, pages 4172–4182, 2023.

Ivan Petrov, Daiheng Gao, Nikolay Chervoniy, Kunlin Liu, Sugasa Marangonda, Chris Umé, Mr. Dpfks, R. P. Luis, Jian Jiang, Sheng Zhang, Pingyu Wu, Bo Zhou, and Weiming Zhang. DeepFaceLab: A simple, flexible and extensible face swapping framework. *ArXiv*, 2020.

Luan Pham, The Huynh Vu, and Tuan Anh Tran. Facial expression recognition using residual masking network. *International Conference on Pattern Recognition*, pages 4513–4519, 2021.

A. J. Piergiovanni and Michael S. Ryoo. AViD dataset: Anonymized videos from diverse countries. In *Conference on Neural Information Processing Systems*, 2020.

Ryan Po, Wang Yifan, Vladislav Golyanik, Kfir Aberman, Jonathan T Barron, Amit H Bermano, Eric R Chan, Tali Dekel, Aleksander Holynski, Angjoo Kanazawa, C Karen Liu, Lingjie Liu, Ben Mildenhall, Matthias Nießner, Bjorn Ommer, Christian Theobalt, Peter Wonka, and Gordon Wetzstein. State of the art on diffusion models for visual computing. *Computer Graphics Forum*, 43, 2023.

Bernardo Pulido-Gaytán, Andrei Nikolaevitch Tchernykh, Jorge M Cortés-Mendoza, Mikhail G Babenko, Gleb I Radchenko, Arutyun I Avetisyan, and Alexander Yu. Drozdov. Privacy-preserving neural networks with homomorphic encryption: Challenges and opportunities. *Peer-to-Peer Networking and Applications*, 14:1666 – 1691, 2021.

Albert Pumarola, Antonio Agudo, Aleix M Martinez, Alberto Sanfeliu, and Francesc Moreno-Noguer. GANimation: Anatomically-aware facial animation from a single image. *European Conference on Computer Vision*, 11214:835–851, 2018.

Delong Qi, Weijun Tan, Qi Yao, and Jingfeng Liu. YOLO5Face: Why reinventing a face detector. In *European Conference on Computer Vision Workshops*, 2021.

Tingting Qiao, Jing Zhang, Duanqing Xu, and Dacheng Tao. MirrorGAN: Learning text-to-image generation by redescription. In *Conference on Computer Vision and Pattern Recognition*, pages 1505–1514, 2019.

Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. *International Conference on Learning Representations*, 2016.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. *International Conference on Machine Learning*, 139:8748–8763, 2021.

Mohammed Gamal Ragab, Said Jadid Abdulkadir, Amgad Muneer, Alawi Alqushaibi, Ebrahim Hamid Hasan Sumiea, Rizwan Qureshi, Safwan Mahmood Al-Selwi, and Hitham Seddiq Alhassan Alhussian. A comprehensive systematic review of YOLO for medical object detection (2018 to 2023). *IEEE Access*, 12:57815–57836, 2024.

Prajit Ramachandran, Barret Zoph, and Quoc V. Le. Searching for activation functions. In *International Conference on Learning Representations*, 2018.

Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. *International Conference on Machine Learning*, 139:8821–8831, 2 2021.

Siddharth Ravi, Pau Climent-Pérez, and Francisco Florez-Revuelta. A review on visual privacy preservation techniques for active and assisted living. *ArXiv*, 2021.

Joseph Redmon and Ali Farhadi. YOLOv3: An incremental improvement. *ArXiv*, 2018.

Joseph Redmon, Santosh Kumar Divvala, Ross B. Girshick, and Ali Farhadi. You Only Look Once: Unified, real-time object detection. *Conference on Computer Vision and Pattern Recognition*, pages 779–788, 2015.

Scott E Reed, Zeynep Akata, Xinchen Yan, Lajanugen Logeswaran, Bernt Schiele, and Honglak Lee. Generative adversarial text to image synthesis. In *International Conference on Machine Learning*, volume 48, pages 1060–1069, 2016.

Shaoqing Ren, Kaiming He, Ross B. Girshick, and Jian Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39:1137–1149, 2015.

Robin Rombach, A Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. *Conference on Computer Vision and Pattern Recognition*, pages 10674–10685, 2022.

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-Net: Convolutional networks for biomedical image segmentation. *Medical Image Computing and Computer-Assisted Intervention*, 2015.

Frank Rosenblatt. The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological review*, 65 6:386–408, 1958.

Nataniel Ruiz, Eunji Chong, and James M Rehg. Fine-grained head pose estimation without keypoints. *Conference on Computer Vision and Pattern Recognition Workshops*, pages 2155–215509, 2018.

David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. Learning representations by back-propagating errors. *Nature*, 323:533–536, 1986.

Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael S Bernstein, Alexander C Berg, and Li Fei-Fei. ImageNet large scale visual recognition challenge. *International Journal of Computer Vision*, 115:211–252, 2015.

Javid Sadr, Izzat N Jarudi, and Pawan Sinha. The role of eyebrows in face recognition. *Perception*, 32: 285 – 293, 2003.

Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L. Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. *Conference on Neural Information Processing Systems*, 35:36479–36494, 2022.

Tim Salimans and Jonathan Ho. Progressive distillation for fast sampling of diffusion models. In *International Conference on Learning Representations*, 2022.

Tim Salimans, Ian J Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training GANs. *ArXiv*, 2016.

Tim Salimans, Andrej Karpathy, Xi Chen, and Diederik P Kingma. PixelCNN++: Improving the PixelCNN with discretized logistic mixture likelihood and other modifications. In *International Conference on Learning Representations*, 2017.

Florian Schroff, Dmitry Kalenichenko, and James Philbin. FaceNet: A unified embedding for face recognition and clustering. *Conference on Computer Vision and Pattern Recognition*, pages 815–823, 2015.

Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, et al. Laion-5b: An open large-scale dataset for training next generation image-text models. *Conference on Neural Information Processing Systems*, 35:25278–25294, 2022.

Vinothini Selvaraju, Nicolai Spicher, Ju Wang, Nagarajan Ganapathy, Joana M Warnecke, Steffen Leonhardt, Swaminathan Ramakrishnan, and Thomas Martin Deserno. Continuous monitoring of vital signs using cameras: A systematic review. *Sensors*, 22, 2022.

Sefik Ilkin Serengil and Alper Ozpinar. Hyperextended lightface: A facial attribute analysis framework. *International Conference on Engineering and Emerging Technologies*, pages 1–4, 2021.

Claude E. Shannon. A mathematical theory of communication. *The Bell System Technical Journal*, 27: 623–656, 1948.

Evan Shelhamer, Jonathan Long, and Trevor Darrell. Fully convolutional networks for semantic segmentation. *Conference on Computer Vision and Pattern Recognition*, pages 3431–3440, 2014.

Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *International Conference on Learning Representations*, 2015.

Pawan Sinha, B. Balas, Yuri Ostrovsky, and Richard Russell. Face recognition by humans: Nineteen results all computer vision researchers should know about. *Proceedings of the IEEE*, 94:1948–1962, 2006.

Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. *International Conference on Machine Learning*, 37:2256–2265, 2 2015.

Gowthami Somepalli, Vasu Singla, Micah Goldblum, Jonas Geiping, and Tom Goldstein. Diffusion art or digital forgery? investigating data replication in diffusion models. *Conference on Computer Vision and Pattern Recognition*, pages 6048–6058, 2022.

Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *International Conference on Learning Representations*, 2021.

Yang Song, Prafulla Dhariwal, Mark Chen, and Ilya Sutskever. Consistency models. In *International Conference on Machine Learning*, volume 202, pages 32211–32252, 2023.

Jost Tobias Springenberg, Alexey Dosovitskiy, Thomas Brox, and Martin A Riedmiller. Striving for simplicity: The all convolutional net. *International Conference on Learning Representations Workshop*, 2015.

Rupesh Kumar Srivastava, Klaus Greff, and Jürgen Schmidhuber. Training very deep networks. In *Conference on Neural Information Processing Systems*, 2015.

Sarah V. Stevenage, Mark S. Nixon, and Kate Vince. Visual analysis of gait as a cue to identity. *Applied Cognitive Psychology*, 13:513–526, 1999.

Chen Sun, Abhinav Shrivastava, Saurabh Singh, and Abhinav Kumar Gupta. Revisiting unreasonable effectiveness of data in deep learning era. *International Conference on Computer Vision*, pages 843–852, 2017.

Yi Sun, Yuheng Chen, Xiaogang Wang, and Xiaoou Tang. Deep learning face representation by joint identification-verification. In *Conference on Neural Information Processing Systems*, pages 1988–1996, 2014a.

Yi Sun, Xiaogang Wang, and Xiaoou Tang. Deeply learned face representations are sparse, selective, and robust. *Conference on Computer Vision and Pattern Recognition*, pages 2892–2900, 2014b.

Yi Sun, Ding Liang, Xiaogang Wang, and Xiaoou Tang. DeepID3: Face recognition with very deep neural networks. *ArXiv*, 2015.

Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott E Reed, Dragomir Anguelov, D. Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. *Conference on Computer Vision and Pattern Recognition*, pages 1–9, 2014.

Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. *Conference on Computer Vision and Pattern Recognition*, pages 2818–2826, 2016.

Christian Szegedy, Sergey Ioffe, Vincent Vanhoucke, and Alexander A. Alemi. Inception-v4, Inception-ResNet and the impact of residual connections on learning. In *AAAI Conference on Artificial Intelligence*, pages 4278–4284, 2017.

Yaniv Taigman, Ming Yang, Marc'Aurelio Ranzato, and Lior Wolf. DeepFace: Closing the gap to human-level performance in face verification. *Conference on Computer Vision and Pattern Recognition*, pages 1701–1708, 2014.

Yong Xuan Tan, Chin-Poo Lee, Mai Neo, Kian-Ming Lim, Jit Yan Lim, and Ali Alqahtani. Recent advances in text-to-image synthesis: Approaches, datasets and future research prospects. *IEEE Access*, 11:88099–88115, 2023.

Philipp Terhörst, Naser Damer, Florian Kirchbuchner, and Arjan Kuijper. Suppressing gender and age in face templates using incremental variable elimination. *International Conference on Biometrics*, pages 1–8, 2019.

Philipp Terhörst, Kevin Riehl, Naser Damer, Peter Rot, Blaz Bortolato, Florian Kirchbuchner, Vitomir Struc, and Arjan Kuijper. PE-MIU: A training-free privacy-enhancing face recognition approach based on minimum information units. *IEEE Access*, 8:93635–93647, 2020.

Patrick J. Tinsley, Adam Czajka, and Patrick J. Flynn. This face does not exist... but it might be yours! identity leakage in generative models. *Conference on Applications of Computer Vision*, pages 1319–1327, 2020.

Patrick J. Tinsley, Adam Czajka, and Patrick J. Flynn. Haven't i seen you before? assessing identity leakage in synthetic irises. *International Joint Conference on Biometrics*, pages 1–9, 2022.

Matthew A. Turk and Alex Pentland. Eigenfaces for recognition. *Journal of Cognitive Neuroscience*, 3:71–86, 1991.

Jasper R. R. Uijlings, Koen E. A. van de Sande, Theo Gevers, and Arnold W. M. Smeulders. Selective search for object recognition. *International Journal of Computer Vision*, 104:154 – 171, 2013.

Ries Uittenbogaard, Clint Sebastian, Julien A Vijverberg, Bas Boom, Dariu M Gavrila, and Peter H. N. de With. Privacy protection in street-view panoramas using depth and multi-view imagery. *Conference on Computer Vision and Pattern Recognition*, pages 10573–10582, 2019.

Dmitry Ulyanov, Andrea Vedaldi, and Victor S. Lempitsky. Improved texture networks: Maximizing quality and diversity in feed-forward stylization and texture synthesis. *Conference on Computer Vision and Pattern Recognition*, pages 4105–4113, 2017.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Conference on Neural Information Processing Systems*, pages 5998–6008, 2017.

Chien-Yao Wang, Hong-Yuan Mark Liao, I-Hau Yeh, Yueh-Hua Wu, Ping-Yang Chen, and Jun-Wei Hsieh. CSPNet: A new backbone that can enhance learning capability of CNN. *Conference on Computer Vision and Pattern Recognition Workshops*, pages 1571–1580, 2019.

Feng Wang, Xiang Xiang, Jian Cheng, and Alan Loddon Yuille. NormFace: L2 hypersphere embedding for face verification. *ACM International Conference on Multimedia*, 2017.

H Wang, Yitong Wang, Zheng Zhou, Xing Ji, Zhifeng Li, Dihong Gong, Jin Zhou, and Wei Liu. CosFace: Large margin cosine loss for deep face recognition. *Conference on Computer Vision and Pattern Recognition*, pages 5265–5274, 2018a.

Jingdong Wang, Ke Sun, Tianheng Cheng, Borui Jiang, Chaorui Deng, Yang Zhao, Dong Liu, Yadong Mu, Mingkui Tan, Xinggang Wang, Wenyu Liu, and Bin Xiao. Deep high-resolution representation learning for visual recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43:3349–3364, 2021a.

Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. High-resolution image synthesis and semantic manipulation with conditional gans. In *Conference on Computer Vision and Pattern Recognition*, pages 8798–8807, 2018b.

Xintao Wang, Ke Yu, Shixiang Wu, Jinjin Gu, Yihao Liu, Chao Dong, Yu Qiao, and Chen Change Loy. ESRGAN: Enhanced super-resolution generative adversarial networks. In *European Conference on Computer Vision Workshops*, volume 11133, pages 63–79, 2018c.

Zhihao Wang, Jian Chen, and Steven C H Hoi. Deep learning for image super-resolution: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43:3365–3387, 2021b.

Ryan Webster, Julien Rabin, Loïc Simon, and Frédéric Jurie. Detecting overfitting of deep generative networks via latent recovery. In *Conference on Computer Vision and Pattern Recognition*, pages 11265–11274, 2019.

Kilian Q. Weinberger and Lawrence K. Saul. Distance metric learning for large margin nearest neighbor classification. In *Conference on Neural Information Processing Systems*, 2005.

Ethan Andrew Wilson, Frédérick Shic, Jenny A. F. Skytta, and Eakta Jain. Practical digital disguises: Leveraging face swaps to protect patient privacy. *ArXiv*, 2022.

Tatjana Wingarz, Marta Gomez-Barrero, Christoph Busch, and Mathias Fischer. Privacy-preserving convolutional neural networks using homomorphic encryption. *International Workshop on Biometrics and Forensics*, pages 1–6, 2022.

Yongkang Wong, Shaokang Chen, Sandra Mau, Conrad Sanderson, and Brian C. Lovell. Patch-based probabilistic image quality assessment for face selection and improved video-based face recognition. *Conference on Computer Vision and Pattern Recognition WORKSHOPS*, pages 74–81, 2011.

Erroll Wood, Tadas Baltruvsaitis, Charlie Hewitt, Sebastian Dziadzio, Matthew Johnson, Virginia Estellers, Thomas Joseph Cashman, and Jamie Shotton. Fake it till you make it: face analysis in the wild using synthetic data alone. *International Conference on Computer Vision*, pages 3661–3671, 2021.

Yuanjun Xiong, Kai Zhu, Dahua Lin, and Xiaoou Tang. Recognize complex events from static images by fusing deep channels. *Conference on Computer Vision and Pattern Recognition*, pages 1600–1609, 2015.

Chugui Xu, Ju Ren, Deyu Zhang, Yaoxue Zhang, Zhan Qin, and Kui Ren. GANobfuscator: Mitigating information leakage under GAN via differential privacy. *IEEE Transactions on Information Forensics and Security*, 14:2358–2371, 2019.

Wanxin Xu, Sen ching S Cheung, and Neelkamal Soares. Affect-preserving privacy protection of video. *International Conference on Image Processing*, pages 158–162, 2015.

Kaiyu Yang, Jacqueline Yau, Li Fei-Fei, Jia Deng, and Olga Russakovsky. A study of face obfuscation in ImageNet. In *International Conference on Machine Learning*, 2021.

Shuo Yang, Ping Luo, Chen Change Loy, and Xiaoou Tang. Wider face: A face detection benchmark. *Conference on Computer Vision and Pattern Recognition*, pages 5525–5533, 2016.

Hu Ye, Jun Zhang, Siyi Liu, Xiao Han, and Wei Yang. IP-Adapter: Text compatible image prompt adapter for text-to-image diffusion models. *ArXiv*, 2023.

Raymond A. Yeh, Chen Chen, Teck-Yian Lim, Alexander G. Schwing, Mark Hasegawa-Johnson, and Minh N. Do. Semantic image inpainting with deep generative models. In *Conference on Computer Vision and Pattern Recognition*, pages 6882–6890, 2017.

Andrew W. Young, Deborah J. Hellawell, and Dennis C. Hay. Configurational information in face perception. *Perception*, 16:747 – 759, 1987.

Matthew D. Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. In *European Conference on Computer Vision*, volume 8689, pages 818–833, 2014.

Bin Zhang, Jian Li, Yabiao Wang, Ying Tai, Chengjie Wang, Jilin Li, Feiyue Huang, Yili Xia, Wenjiang Pei, and Rongrong Ji. ASFD: Automatic and scalable face detector. *ACM International Conference on Multimedia*, 2021.

Kaipeng Zhang, Zhanpeng Zhang, Zhifeng Li, and Yu Qiao. Joint face detection and alignment using multitask cascaded convolutional networks. *IEEE Signal Processing Letters*, 23:1499–1503, 2016.

Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. *International Conference on Computer Vision*, pages 3813–3824, 2023.

Wenchao Zhang, S. Shan, Wen Gao, Xilin Chen, and Hongming Zhang. Local gabor binary pattern histogram sequence (LGBPHS): a novel non-statistical model for face representation and recognition. *International Conference on Computer Vision*, 1:786–791 Vol. 1, 2005.

Zhaohui Zheng, Ping Wang, Wei Liu, Jinze Li, Rongguang Ye, and Dongwei Ren. Distance-IoU Loss: Faster and better learning for bounding box regression. In *AAAI Conference on Artificial Intelligence*, 2020.

Linqi Zhou, Yilun Du, and Jiajun Wu. 3D shape generation and completion through point-voxel diffusion. In *International Conference on Computer Vision*, pages 5806–5815, 2021.

C. Lawrence Zitnick and Piotr Dollár. Edge boxes: Locating object proposals from edges. In *European Conference on Computer Vision*, 2014.

Appendix A

Notation

Here, we clarify the mathematical notation used in this thesis. When possible, we have chosen unique parameters for our equations. However, there are some cases where the usage of a certain parameter is so common that using a different one for the sake of disambiguation would cause confusion.

Parameter	Meaning	Where
α	Auxiliary parameter used to derive the loss function for diffusion models $\alpha_t = 1 - \beta_t$ with the variance β. Can also be interpreted as the amount of signal preserved at timestep t	Section 2.5.2
β	Variance of noise	Section 2.5.2
δ	Margin in triplet loss. α is more common in the literature on triplet loss and contrastive learning [Schroff et al., 2015].	Section 2.4
ϵ	Noise typically sampled from a Gaussian distribution	Section 2.5.2
η	Learning rate	Section 2.1
κ	Scale factor for hypersphere radius in face face recognition. More commonly denoted as s [Deng et al., 2019; Wang et al., 2018a]	Section 2.4
λ	Weighting factor. λ_{pro} is typically referred to as α [Karras et al., 2018].	Sections 2.3 and 2.5.1

μ	Mean	Sections 2.5.2 and 3.2.2
ϕ	Angle between Weights W of the final classification layer and representation r in face recognition. In the literature [Deng et al., 2019; Liu et al., 2017] this is typically denoted as θ.	Section 2.4
σ	standard deviation	Section 3.2.2
Σ	Variance	Sections 2.5.2 and 3.2.2
θ	All learnable parameters $\{W_i, b_i\}_{i=1}^{I}$ of a neural network	Section 2.1
b	Learnable bias parameter of a neural network	Section 2.1
c	Index describing the class label in classification, detection or face recognition tasks	Sections 2.2 to 2.4
c	Conditioning input for GANs or Diffusion models	Sections 2.5.1 and 2.5.2
d	Channel dimension of an image or a feature map	Sections 2.2 and 2.3
h	Height of an image or a feature map	Section 2.3
k	Dimension of a convolutional kernel	Section 2.2
K	Confidence predicted by YOLO. More commonly denoted as C [Redmon et al., 2015]	Section 2.3
m	Angular margin	Section 2.4
M	Number of samples in a mini-batch	Section 2.5.1
s	Guidance (cfg) scale	Section 2.5.2
s	Stride for CNN kernels	Section 2.2
S	Number of grid cells in one dimension. $S \times S$ is the total number in two dimensions.	Section 2.3
t	Timestep index	Section 2.5.2
T	Total number of timesteps	Section 2.5.2

w	Width of an image or feature map	Section 2.3
w	Weight vector describing the weights of a perceptron or single neuron	Section 2.1
W	Weight matrix describing the weights of a neural network	Sections 2.1 and 2.4
W	Intermediate latent space in StyleGAN	Section 2.5.1
x	Spatial coordinate	Section 2.3
x	Input to a neural network	Section 2.4
y	Spatial coordinate	Section 2.3
y	Ground truth of a data set	Section 2.1
z	Noise	Section 2.5.1

Table A.2: Mathematical notation used in this thesis.

Notation	**Explanation**
$\mathcal{N}(x;\mu,\Sigma)$	Variable x follows the Gaussian distribution with mean μ and covariance Σ
$\mathbb{E}$	Expected value
$z \sim p_z$	Sample z from distribution p_z
$\mathbf{I}$	Identity matrix
$[x]_+$	max(0,x)
W^T	Transposed matrix W
Tr(A)	Trace of matrix A

Table A.3: Relevant acronyms used in this thesis.

Acronym	**Explanation**
ANN	Artificial Neural Network
CIoU	Complete Intersection over Union
CNNs	Convolutional Neural Networks
Conv.	Convolutional layer
CSP	Cross-Stage Partial connections
DFL	Distribution Focal Loss
DPM	Deformable Part Model
DSFD	Dual Shot Face Detector
FC	Fully Connected Neural Network
FEM	Feature Enhancement Module
FPN	Feature Pyramid Network
FSL	First Shot Loss
GAN	Generative Adversarial Network
ILSVRC	ImageNet Large Scale Visual Recognition Challenge
LDM	Latent Diffusion Model
MAE	Mean Absolute Error
MSE	Mean Squared Error
NMS	Non-Maximum Suppression
PAL	Progressive Anchor Loss
PCA	Principal Component Analysis
PS	Perceptual Sensitivity
ReLU	Rectified Linear Unit
SGD	Stochastic Gradient Descent
SIFT	Scale-Invariant Feature Transform
SSL	Second Shot Loss
SPP	Spatial Pyramid Pooling
SPPF	Spatial Pyramid Pooling Fast
SVM	Support Vector Machine
YOLO	You Only Look Once