Designing and evaluating anonymization techniques for images and relational data streams via Machine Learning approaches at BMW Group

Jimmy Tekli

UBFC
UNIVERSITÉ
BOURGOGNE FRANCHE-COMTÉ

UFC
UNIVERSITÉ
DE FRANCHE-COMTÉ

THÈSE DE DOCTORAT

DE L'ÉTABLISSEMENT UNIVERSITÉ BOURGOGNE FRANCHE-COMTÉ

PRÉPARÉE À L'UNIVERSITÉ DE FRANCHE-COMTÉ

École doctorale n°37

Sciences Pour l'Ingénieur et Microtechniques

Doctorat d'Informatique

par

JIMMY TEKLI

Designing and evaluating anonymization techniques for images and relational data streams via Machine Learning approaches at BMW Group

Conception et évaluation de techniques d'anonymisation des images et des flux de données relationnels via des approches d'apprentissage automatique à BMW Group.

Thèse présentée et soutenue à Belfort, le 17 Décembre 2021

Composition du Jury :

ELIZONDO DAVID	Professeur à Université de Monfort - U.K.	Rapporteur
CHBEIR RICHARD	Professeur à Université de Pau et des Pays de l'Adour - France	Rapporteur
CLIFTON CHRIS	Professeur à Université de Purdue - USA	Examinateur
NASSAR MOHAMED	Professeur à Université de New Haven - USA	Examinateur
BRUNIE LIONEL	Professeur à l'INSA Lyon - France	Examinateur
COUTURIER RAPHAËL	Professeur à l'UBFC - France	Directeur de thèse
AL BOUNA BECHARA	Professeur à l'Université Antonine - Liban	Codirecteur de thèse

N° | X | X | X |

ACKNOWLEDGEMENTS

I would like to thank and express my deepest gratitude to my co-supervisors Dr. **Albouna Bechara** and Dr. **Couturier Raphaël** first for trusting me, giving me the opportunity to conduct my PhD under their supervision and for their guidance and engagement during the last 4 years. Second, I would like to thank my supervisor at BMW Group Mr. **Kamradt Marc** for providing all the resources (financially and technically) that I needed to accomplish my PhD as well as the opportunity to work on several R&D projects with numerous students which helped me a lot throughout my thesis. Third, I thank my colleagues Dr. **Nassif Jimmy** and Mr. **El Asmar Boulos** who were a major support throughout the first year of my thesis at BMW Group. I would also like to thank my colleagues Dr. **Awaad Nancy**, Mr. **El Chami Zahi** and my friend **Raad Ralph** for their continuous support. As for my family back in Lebanon, words cannot express my love and gratitude for every single member. **Mom and Dad**, I wouldn't be here if it wasn't for you two. I love you from the bottom of my heart. Your presence in my life is a true blessing which I cherish and thank God for each day. As for my brothers **Joe** and **Gilbert**, both have been a true inspiration on so many levels (although they make it hard sometimes). Also, would like to specially thank **Gilbert** who was there for me during the last 2 years, his guidance as well as his support were essential to me accomplishing my thesis. As for my sisters **Hannoud** and **Sannoud**, their presence, positive energy and continuous support was and is always a huge boost. Also, I cannot thank them enough for bringing the two rascals **Kavvoun** and **Ausi** into my life. Finally, I want to thank my aunts/uncles and cousins **Abou greg, Petit Joe, Piero, Tartoura, Lallous, Abou Zouz, Dan Dan, Mano, Zoun and Raf Raf** for their lifetime support. I just love you guys and hope to see u soon. Also, I would like to thank my girlfriend **Natalie** for being there for me during the last year of my thesis and for re-reading some parts of it:)!! Without forgetting our loved ones, especially my guardian angel **Martha Aoun**, hovering above and watching over us. Thank you, God, for all these blessings.

CONTENTS

I

INTRODUCTION

1

INTRODUCTION

1.1/ INDUSTRY 4.0

We live in an era where the world is more connected than ever before and everything is digitized from smartphones, smart vehicles to smart homes and smart cities. This technological shift propagated as well throughout the industrial world and is at the heart of the 4th Generational Industrial revolution (Industry 4.0) [Gilchrist, 2016, Greengard, 2015]. From supply chain optimization to autonomous driving vehicles, manufacturing units and automotive companies are increasingly integrating smart sensors, smart cameras, smart robots and many more Internet Of Things (IoT) devices to further improve the efficiency of their processes and the quality of their products.

1.2/ DATA HETEROGENEITY

Despite the heterogeneity of these IoT devices, they all have one thing in common: they generate vast amount of data. Furthermore, the data generated is as heterogeneous as the smart devices themselves. For instance, IoT devices such as smart sensors and cameras installed in smart vehicles or throughout a smart factory generate different data types including:

- **Digital image:** consists of a set of pixels organized in a form of a grid. Each pixel value denotes its brightness level (e.g. gray-scale image) or its color intensity (e.g. RGB image). The collection of these pixels constitutes the image's features such as the image's edges, corners, ridges, etc.
- **Digital video**: consists of a series of digital images combined and displayed in succession.
- **Structured relational data:** consists of a table with different attributes and tuples containing either categorical or quantitative values.

1.3/ DATA CYCLE

Upon data generation, three main actors play major roles throughout the collection, release, and usage of the data :(i) data owners, (ii) data collectors and (iii) data consumers. Data owners own the data and decide what will happen to it after generation. Whereas data collectors (e.g., data engineers) collect and release the data in the following format:

- **Static bulk dataset**: all data points are grouped into one fixed size dataset that is released once for processing purposes.
- **Sequential/incremental dataset**: a group of data points is collected into one subset at each time interval and released for processing purposes. In other words, the released dataset is a combination of subsets collected at different time intervals.
- **Data stream**: a new data point is generated, collected and released at each time instant for processing purposes.

Last but not least, data consumers (e.g., data scientists in other organizational units within the same company or 3rd party providers) extract relevant insights from the released data via descriptive reporting, correlation or predictive analytics. Several concepts such as supervised Machine Learning [Muhammad et al., 2015] (e.g. Deep Learning [Bengio, 2009]), unsupervised Machine Learning [Grira et al., 2004] (e.g. clustering) or Reinforcement Learning (RL) [Kaelbling et al., 1996, Arulkumaran et al., 2017] are usually employed by the data consumers when performing the reporting/analytical tasks either in a centralized, distributed [Verbraeken et al., 2020a] or collaborative learning process (e.g., Federated Learning [McMahan et al., 2017, Yang et al., 2019a]). In the following, we briefly elaborate on the most relevant concepts to our study.

- **Machine Learning (ML)** [Qiu et al., 2016]: is an application of artificial intelligence (AI) that provides machines the ability to automatically learn and predict certain outcomes without being explicitly programmed to do so. ML approaches accomplish that by iterating several times over data points and are usually classified into supervised and unsupervised. Supervised approaches (e.g. Deep learning) learn and improve from labeled data points whereas unsupervised ML (e.g. clustering) learn and discover patterns from unlabeled data.
- **Deep Learning (DL)** [Bengio, 2009]: Artificial Neural Network (ANN) [Haykin, 2010] is a computational nonlinear model inspired by the biological systems in information processing. It consists of artificial neurons (a.k.a. perceptrons) interconnected to form three distinct layers: (i) input layer, (ii) hidden layer and (iii) output layer. Artificial neural networks with more than one hidden layer are called Deep Neural

Networks (a.k.a. Deep Learning). DL techniques mainly employ supervised feature learning to map input raw data (e.g. digital images) to a specific output (e.g., class, bounding box, etc...). In other words, a deep neural network "learns" to extract relevant features in an end-to-end manner via input/label pairs. Please refer to Chapter 2 for more details regarding DL and the learning process.

- **Clustering** [Grira et al., 2004]: is the task of grouping data points into clusters such that data points within the same cluster have similar characteristics when compared to data points in other clusters. This grouping can be achieved in an unsupervised manner. Some of known clustering algorithms are K-means [Hartigan et al., 1979] and hierarchical clustering [Johnson, 1967].
- **Federated Learning (FL)** [McMahan et al., 2017, Yang et al., 2019a]: is a ML setting where multiple clients (e.g., data owners) collaborate in solving a machine learning problem under the coordination of a central server/coordinator. Each client's raw data is stored locally without being exchanged nor transferred to the central server; instead, the model's parameters are shared/aggregated and used to achieve the learning objective. Please refer to Chapter 5 for additional details regarding the FL concept.

1.4/ PRIVACY: A NECESSITY

Generated data, whether it is static, sequential or a data stream, might contain information relating to an identified or identifiable data subject, i.e. personal data [Jensen et al., 2019]. As defined in the Cambridge English Dictionary[1], privacy is "someone's right to keep their personal matters and relationships secret". Individual's privacy and anonymity is more critical in our data-driven world due to the vast amount of data being generated, released and processed daily. Hence, we talk about **data privacy**.

1.4.1/ DATA PRIVACY

Data privacy governs how data is collected, released and used. Several data protection regulations have been introduced by governments in the last couple of years such as the General Data Protection Regulation (GDPR) in Europe [European Parliament, 2018b] and the California Consumer Privacy Act (CCPA) [Legislature, 2018] in the United States to protect the personal data and the privacy of the data subjects. Article 5 of the GDPR lists seven principles to consider when processing[2] personal data

[1] https://dictionary.cambridge.org/dictionary/english/privacy

[2] processing includes data collection, organisation, structuring, storage, alteration, consultation, use, communication, combination, restriction, erasure or destruction.

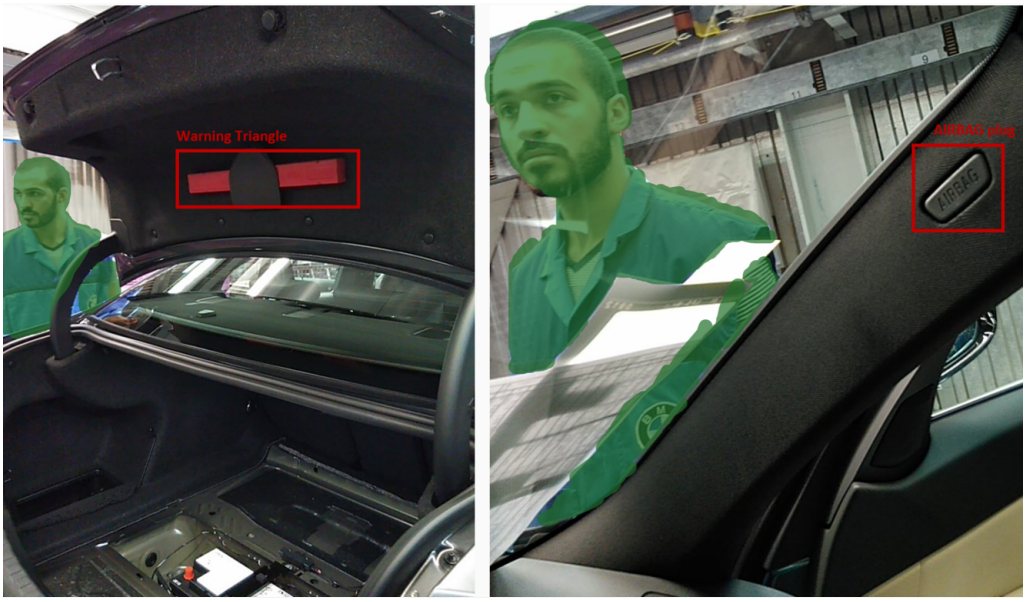
[European Parliament, 2018c]. In short, the seven principles are:

- Lawfulness, fairness and transparency
- Purpose limitation
- Data minimisation
- Accuracy
- Storage limitation
- Integrity and confidentiality (security)
- Accountability

Organizations are obliged to respect and enforce these seven principles to process personal data in a compliant way. Nonetheless, we elaborate on the third principle as it is the most relevant to our study. “Data minimization” is defined in Article 5(1)(c) as the following: ”Personal data shall be adequate, relevant and limited to what is necessary in relation to the purposes for which they are processed” [European Parliament, 2018a].

In the following, we present an example scenario at a BMW Group manufacturing unit to demonstrate when the question of data privacy arises and explain how the ”data minimization” principle is relevant.

Example Scenario at BMW Group: Throughout the production process of a vehicle, workers perform multiple quality checks around the assembly lines by inspecting different parts and checking for faulty manufacturing (such as missing warning triangles, airbag stickers or scratches as seen in Figure 1.1). ML and more specifically DL algorithms trained for Computer Vision (CV) tasks (e.g., object detection [Liu et al., 2016], image classification [Russakovsky et al., 2015a]...) are deployed to assist plant workers with the vehicle’s inspections to reduce the quality check cycles. On the one hand, training these DL models requires capturing, storing and more specifically labeling large amounts of images by ’human labelers’ [Ayle et al., 2020]. On the other hand, these captured images might contain personal/sensitive information such as workers’ faces, workers’ belongings, or even name tags. These sensitive features are not relevant for labeling nor for training the DL models to detect vehicle parts. Hence, using and labeling these images without removing these identifying features is a breach to the ”data minimization” principle.



(a) (b)

Figure 1.1: Automated Quality Checks via Object Detection techniques in order to check that (a) Warning triangles and (b) Airbag Plugs are installed correctly on the corresponding vehicles in production. The worker's identifying features are highlighted in green.

1.4.2/ Data anonymization

As defined in [European Parliament, 2019, European Parliament, 2018b], data anonymization is the process of creating anonymous information, namely information which does not relate to an identified or identifiable natural person in such a manner that the data subject is not or no longer identifiable. Numerous anonymization mechanisms/techniques have been proposed in the literature for each data type. For instance, perturbative [Dwork, 2008] and non-perturbative [Samarati et al., 1998] mechanisms were proposed to guarantee the anonymity of the data subjects in structured relational datasets. Similarly, many anonymization[3] techniques (a.k.a. obfuscation techniques in the context of images) such as pixelating, blurring or masking have been used to protect/hide personal/sensitive (i.e., identifying) features in images by modifying the images' pixels [Hill et al., 2016b].

1.4.3/ Privacy attacks

All anonymization techniques, whatever the data type, take into consideration the trade-off between privacy and utility. It is a trade-off that is highly required to keep the dataset suitable for analysis while preserving the data subject's anonymity. How-

[3]Throughout the rest of this study, we will use the terms obfuscation and anonymization interchangeably in the context of digital images.

ever, sometimes it keeps anonymization vulnerable and unable to cope with all sort of attacks [Wong et al., 2011a, Cormode et al., 2010, Kifer, 2009a, Bouna et al., 2015, McPherson et al., 2016]. As defined in [Do et al., 2018] within the field of security, an adversary refers to an attacker, often with malicious intents, that undertakes an attack on a secure system to prevent or disrupt its proper operation. The adversary is presented as a three-component model having a goal, assumptions (i.e., knowledge) and capabilities (c.f. Figure 1.2).

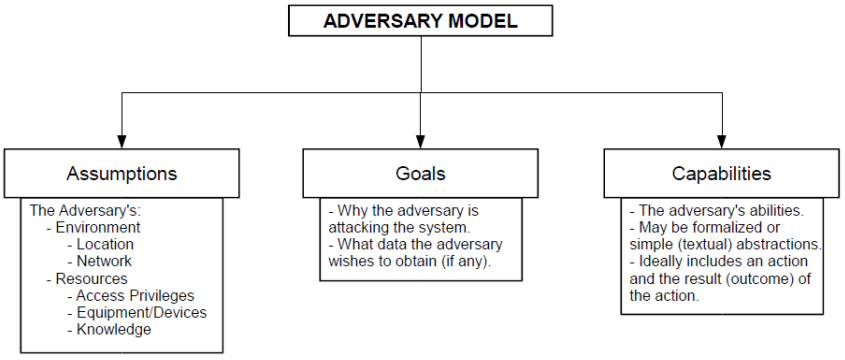


Figure 1.2: The three components of the adversary model defined in [Do et al., 2018]

As the European Union Agency for Cybersecurity (ENISA) mentioned in their technical report published in 2019 [Jensen et al., 2019], anonymization techniques and privacy models should always be re-investigated as the adversaries (adversarial models) are evolving and becoming more challenging. Several privacy breaches proved throughout the years that anonymization techniques should always be revisited: starting from the Sweeny study back in 2000 [Sweeney, 2000] where the author linked de-identified patient-specific medical data to a voters' list and showed that 87.1% of the United States population was uniquely identifiable from the combination of their Date of Birth (DoB), Sex and Zip code, to the AOL [Barbaro et al., 2006] and Netflix cases [Narayanan et al., 2006] in 2006. Please refer to [Awad, 2020] for a detailed description of these attacks. Below, we present two more recent privacy breaches, more specifically in the context of relational datasets and images:

- **The case of Taxa 4x35** [Board, 2019]: Taxa 4×35 is a Danish taxi service that allows its users to hail cabs in Copenhagen with an application. Similar to any cab-driving company, taxa collects data including the customer's name, telephone number, the date of the trip, the payment methods... Taxa anonymized the data by deleting the names associated with the trip records from their database. After adapting/applying the GDPR law and principles in 2018, the Danish authorities, i.e. Datailsynet, realized that this anonymization technique is inadequate and that even after deleting the customer's name, taxa still had enough information (e.g.

customer's phone number) to re-identify each individual in the dataset.

- **Breaching privacy in obfuscated images**: the authors in [McPherson et al., 2016], demonstrated that modern image recognition approaches (i.e. DL models) can be employed to recover hidden information from obfuscated protected images [McPherson et al., 2016]. The adversary successfully identified obfuscated faces and objects by training image recognition networks on obfuscated images (faces [Ng et al., 2014], digits [LeCun, 1998] and objects [Krizhevsky et al., 2009]).

1.5/ PROBLEM AND CONTRIBUTIONS

Privacy regulations [European Parliament, 2018b] compel data-driven companies to guarantee a level of anonymization that requires "irreversibility preventing identification of the data subject", taking into account all the means "reasonably likely to be used" for identification. In other words, when applying an anonymization technique or a privacy model, one should carefully study its resiliency and robustness against adversaries, e.g., the motivated intruder's test proposed in the ICO code of conduct[4]. Therefore throughout our thesis, we (i) propose and implement several anonymization techniques and tools in the context of images and relational data streams and (ii) assess the robustness of these techniques by simulating adversaries with different knowledge and several attacking capabilities. More specifically, our contributions can be summarized as the following:

1. In the first contribution, we design and implement an anonymization tool that localizes sensitive information in images/videos via DL-based techniques and obfuscates it accordingly. This chapter was published as a public GitHub Repository [Tekli et al., 2021], as a press release by BMW Group [Hatzel, 2021] and as a white paper by Intel Cooperation [Intel, 2021].

2. In the second contribution, we study the robustness of obfuscation techniques in the context of images, more specifically facial images. We propose a recommendation framework that evaluates the robustness of image obfuscation techniques and recommends the most resilient obfuscation against adversaries executing DL-assisted attacks. We embed and adapt the three-component model proposed in [Do et al., 2018] to the facial image obfuscation context. We also study thoroughly the privacy breaches in three threat levels with regard to the adversary's knowledge. This chapter was published partially in the 17th International Conference on

[4] https://ico.org.uk/for-organisations/guide-to-data-protection/guide-to-the-general-data-protection-regulation-gdpr/accountability-and-governance/codes-of-conduct/

Privacy, Security and Trust (PST) in 2019 [Tekli et al., 2019] and is currently under peer review in a scientific journal [Tekli et al., NDb]

3. In the third contribution, we empirically demonstrate how adversaries can remedy their lack of knowledge and leverage their attacking capabilities, against obfuscated facial images, by collaborating via FL. We define and study 7 collective threat levels based on the background knowledge of the adversaries and the sharing of their knowledge. This chapter is currently under peer review in a scientific journal [Tekli et al., NDa].

4. In the final contribution, we study the correlation problem after applying anatomy [Xiao et al., 2006a] to a relational transactional data stream. We demonstrate the privacy breaches by simulating two threat levels with regard to the adversary's knowledge. Then, we define privacy properties required to bind the correlation in a data stream and we propose a novel clustering approach to enforce the aforementioned privacy properties by anonymizing incoming tuples on the fly. This chapter was published in the International Conference on Information Security Practice and Experience ISPEC 2018 [Tekli et al., 2018].

1.6/ OUTLINE OF THE THESIS DISSERTATION

This thesis report is split in two main parts. In Part II, we present preliminaries and background information regarding data privacy in the context of digital images and relational datasets. In addition, we discuss briefly the basic concepts of DL and its applications in CV. In Part III, we elaborate on our 4 contributions respectively in Chapters 3, 4, 5 and 6. In Chapter 3, we design and implement an obfuscation tool that localizes and obfuscates sensitive/identifying features in images/videos. In Chapter 4, we propose a generic and scalable framework to evaluate and recommend the most robust obfuscation techniques for face images. The framework reconstructs/recognizes obfuscated faces via DL-assisted attacks, evaluates the restoration/recognition via different metrics and recommends the most robust obfuscation with regard to each metric. In Chapter 5, we empirically demonstrate that FL can be used as a collaborative attack/adversarial strategy to (i) remedy the lack of background knowledge, (ii) leverage the attacking capabilities of an adversary and increase the privacy breaches. As for Chapter 6, we define new privacy properties to address the correlation problem in the anonymization of a transactional data stream. We propose a clustering-based technique to enforce the privacy properties. Last but not least, we conclude our report in Part IV and give some perspectives.

II

PRELIMINARIES

2

Background And Preliminaries

2.1/ Introduction

This chapter serves as a preliminary discussion to Part III. We mainly present basic privacy concepts regarding two data types: (i) digital images and (ii) structured relational data. First, we define in Section 2.2 a digital image, the feature extraction process and we briefly discuss the basic concepts of DL and its CV applications. Furthermore, we present in Section 2.3 obfuscation techniques in the context of images and attacks employed against these techniques. In Section 2.4, we discuss several privacy preserving mechanisms and privacy models proposed in the literature in the context of structured relational datasets and we describe the correlation problem in the context of a transactional dataset. Sections 2.2 and 2.3 are relevant to chapters 3, 4 and 5 whereas Section 2.4 is relevant to Chapter 6. Last but not least, the reader can refer to Chapters 3,4,5 and 6 for a deep dive into the related works regarding each contribution.

2.2/ Deep Learning for Computer Vision applications

In the following section, we define a digital image and we elaborate on how to extract relevant features from it. Second, we present the basic concepts of an artificial neural network and of a convolution neural network. Last but not least, we enumerate several CV tasks that we employed throughout our study using DL models.

2.2.1/ Digital images

As mentioned in Chapter 1, a digital image consists of a set of pixels organized in a form of a grid. Each pixel value denotes its brightness level (e.g. gray-scale image) or its color intensity (e.g. RGB image). The collection of these pixels constitutes the image's features such as the image's edges, corners, ridges, etc. CV applications allow machines to

visualize, perceive and semantically understand their environments by extracting relevant features from digital images.

2.2.2/ FEATURE EXTRACTION

Images feature extraction can be done either via (i) hand-crafted feature engineering or (ii) feature learning [O'Mahony et al., 2019]. Hand-crafted feature engineering refers to the process of employing/customizing specific techniques/algorithms (edge detection algorithms, feature descriptors SIFT[1] [Karami et al., 2017] and SURF[2] [Bay et al., 2006], HOG[3] [Dalal et al., 2005]...) to extract specific features from an image such as edges, corners, ridges... Feature learning allows a system to automatically discover the representations needed for feature detection or classification from raw images without deliberately implementing different extraction techniques via a pipeline.

Early on, CV applications were achievable by a 2-steps process: (i) applying a hand-crafted feature extraction algorithm (e.g. feature descriptors SIFT [Karami et al., 2017] and SURF [Bay et al., 2006], HOG [Dalal et al., 2005]...) followed by (ii) training a traditional machine learning model on the extracted features [Viola et al., 2001].

Throughout the last two decades, artificial neural networks and more specifically deep neural networks (DNNs) [Bengio, 2009] (i.e. Deep Learning DL) outperformed the traditional 2-steps process in terms of accuracy and inference time [O'Mahony et al., 2019]. DNNs and more specifically Deep Convolutional Neural Networks (DCNNs) "learn" to extract relevant features from input raw images in an end-to-end manner (c.f. Figure 2.1).

In the following section, we give a quick overview regarding the basic concepts of artificial neural networks, how the learning process is achieved and how it can be improved. In addition, we present swiftly the basic components of a CNN and the different CV applications that we employ throughout this study.

2.2.3/ ARTIFICIAL NEURAL NETWORK

2.2.3.1/ ARCHITECTURE

An Artificial Neural Network [Haykin, 2010] is a computational nonlinear model inspired by the biological systems in information processing. It consists of artificial neurons (a.k.a. perceptrons) interconnected to form three distinct layers: (i) input layer, (ii) hidden layer and (iii) output layer (c.f. Figure 2.2). Raw data is fed to the network via the input layer. The input layer then forwards the data to one or more hidden layers where the actual

[1] SIFT: Scale Invariant Feature Transform
[2] SURF: Speeded Up Robust Features
[3] histogram of oriented gradients

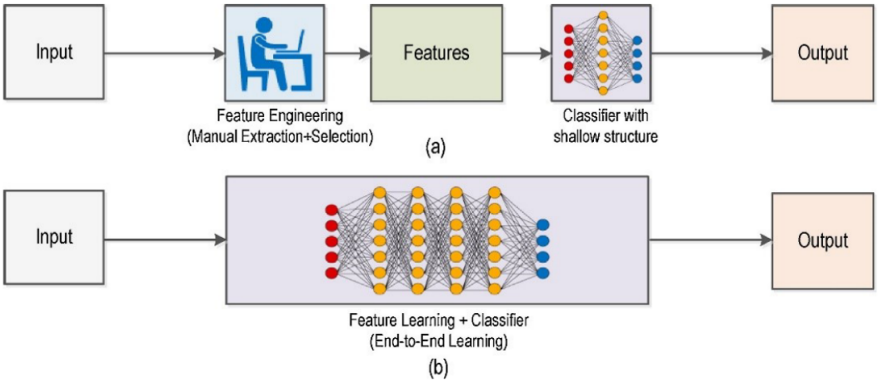


Figure 2.1: (a) Traditional Computer Vision Workflow vs. (b) Deep Learning workflow. Image taken from [O'Mahony et al., 2019]

processing takes place via weighted connections before being sent to the output layer. Artificial neural networks with many hidden layers are called Deep Neural Networks (a.k.a. DNNs). Deep learning techniques mainly employ supervised feature learning to map input raw data (e.g. digital images) to a specific output (e.g., class, bounding box, etc. . .). In other words, a deep neural network "learns" to extract relevant features in an end-to-end manner via input/label pairs. In the following section, we briefly explain the learning process.

2.2.3.2/ LEARNING PROCESS

The learning process can be split into two main steps: (i) feed-forward and (ii) backward pass.

- **Feed-forward pass**: upon receipt of input raw data, a DNN sequentially passes the feature data from one layer to the next. This process is known as feed-forward passing. More relevant (i.e. high level) features are extracted at deeper layers. As seen in Figure 2.3, each neuron receives weighted inputs from the previous layer. These weighted inputs are summed and fed to an internal activation function. The output of the neuron mainly depends on the type of the activation function: *for instance, if the RELU function [Agarap, 2018] is used, then the output of the neuron would be zero if the weighted sum is negative*. Other activation functions can be employed as well such as Logistic (Sigmoid) and hyperbolic Tangent (Tanh) functions.

- **Backward pass**: as mentioned before, DL techniques often employ supervised feature learning therefore, each given input has a label. Based on the raw data fed to the input layer, the network generates a certain output. This output is then compared with the input's label to calculate an error value. This error value is used

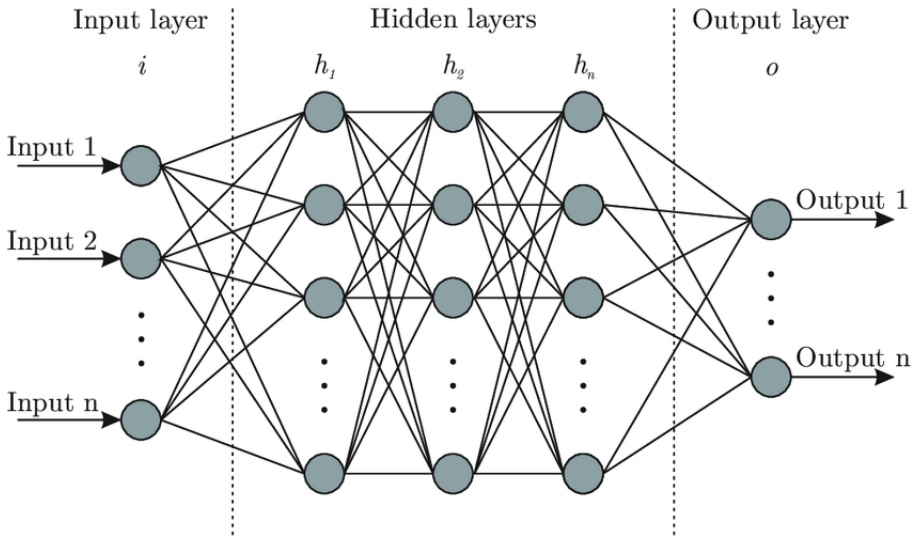


Figure 2.2: Deep Neural network architecture.

to compute the loss function and update the layers' weights via backpropagation. The exact type of loss function depends on the nature of the model, but it is essentially a tool for evaluating the performance of a model on some given data. The objective is to minimize the loss function and compute the combination of weight values to reach that objective. Therefore, both the feed-forward and the backward passes are repeated numerous times over the training data (input/labels) to update the weight values and minimize this loss function. This repeated workflow is known as 'learning'.

2.2.3.3/ WEIGHTS OF A NEURAL NETWORK

In addition to the neural network's architecture, the weights assigned to each connection between the different layers play a vital role in minimizing the loss function. They basically represent the 'knowledge' of the network. As mentioned in [Georgevici et al., 2019], the weights are analogues to coefficients in a traditional statistical model and are also known as the model's parameters. Compared to a large multivariable statistical model which might contain fewer than 50 coefficients, even small DNN can have many thousands of weights, while large recurrent or convolutional networks often have many millions. We denote in our report, both the network's architecture and the corresponding weights as a "DL model".

2.2.3.4/ TRAINING AND TEST SETS

Like any machine learning model, training a neural network requires partitioning the dataset into two sets: training and test sets. On the one hand, the network uses the

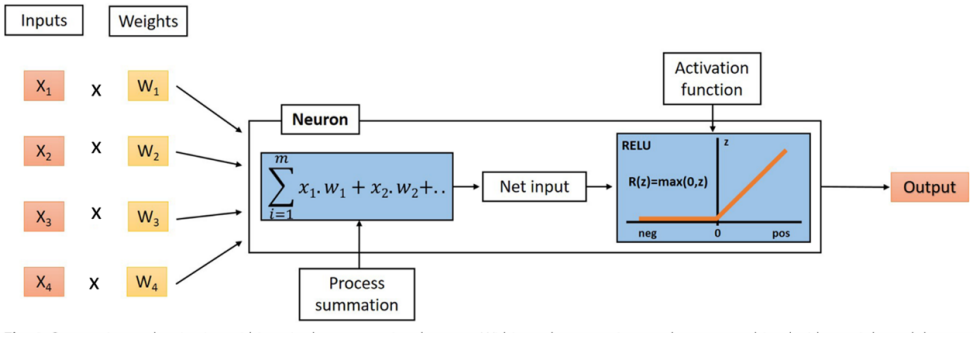


Figure 2.3: Summation and activation within a single perceptron. Image taken from [Georgevici et al., 2019]

training set examples to calculate the loss function and update its weights accordingly (i.e. to fit the parameters). On the other hand, the test set provides an "unbiased" evaluation of the already trained model. In other words, it is used to check if the learned weights generalize well on unseen data.

2.2.3.5/ HYPERPARAMETERS

The hyperparameters are variables that determine (i) the network's architecture and (ii) the network's training configuration. The number of hidden layers, the regularization technique (e.g., dropout), the network's weight initialization (e.g. Xavier initialization [Glorot et al., 2010]) in addition to choosing the activation function: all these are hyperparameters related to the networks' architecture. Whereas, the hyperparameters related to the network's training configuration affect basically the training process. In the following, we present swiftly the hyperparameters affecting the network's training configuration as we employed/modified them frequently when training DL models throughout this study.

- **Learning rate**: indicates at which pace the weights get updated. It can be fixed or adaptively changed during the training process (i.e. adaptive algorithm Adam [Kingma et al., 2014]). Adaptive techniques tend to reach better results in comparison with choosing a fixed learning rate throughout the entire training process.
- **Number of epochs**: is the number of times the whole training data is fed to the network while training.
- **Batch size**: is the number of data samples fed to the network after which the parameter/weight update occurs.

Carefully choosing these hyperparameters might lead to a more accurate DL model. Furthermore, additional steps can be considered when the training dataset is not big enough

to train an accurate DL model.

- **Data augmentation** [Shorten et al., 2019]: DL models usually need a lot of data to be properly trained. Therefore, creating additional data points from existing ones using data augmentation techniques might enhance the training process. *For instance, employing rotation or flipping techniques in the context of images lead to additional images that are not identical to the original one.*
- **Transfer learning** [Weiss et al., 2016, Shao et al., 2014]: as mentioned before, training a deep learning model requires lots of data and more importantly a long time. However, one can apply already pre-trained weights to the neural network architecture to fasten the training process. These pre-trained weights are usually trained on huge datasets with numerous features.

2.2.3.6/ CONVOLUTIONAL NEURAL NETWORKS

There exist different deep neural network architectures that target problems in different domains such as Convolutional Neural Networks (CNN), Deep Belief Networks (DBN), Recurrent Neural Networks (RNN)... In the following, we present briefly CNNs as they are the most widely used when it comes to digital images.

In the last decade, Convolutional Neural Networks (CNN) [LeCun et al., 1998] have become one of the most popular techniques in computer vision. CNNs outperformed traditional ML approaches in several CV tasks such as image classification [Krizhevsky et al., 2012], object detection [Jiao et al., 2019], image segmentation [Minaee et al., 2021, Chen et al., 2017] and image restoration [Koh et al., 2021, Pushpalwar et al., 2016, Nasrollahi et al., 2014]. This is mainly due to two main reasons: (i) the development of efficient computing hardware (e.g., Graphical Processing Units GPUs) and (ii) the large amount of publicly available datasets [Deng et al., 2009, Lin et al., 2014, Cordts et al., 2016].

A CNN typically consists of three main layers, (i) a convolutional layer, (ii) a pooling layer and (iii) Fully connected layer:

- **The Convolutional layer:** uses filters (i.e. kernels) that perform convolution operations as it is scanning the input image with respect to its dimensions. The resulting output is called feature map or activation map (c.f. Figure 2.4).

- **The Pooling Layer**: is a down-sampling operation, typically applied over a feature map. The Pooling layer can down-sample the feature map either by considering the maximum or the average value as seen in Figure 2.5.

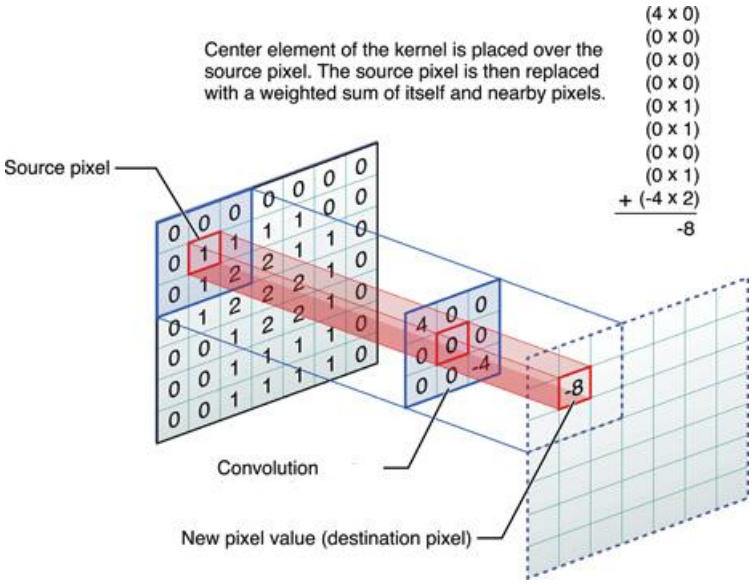


Figure 2.4: Convolution process. Image taken from [Srihari,]

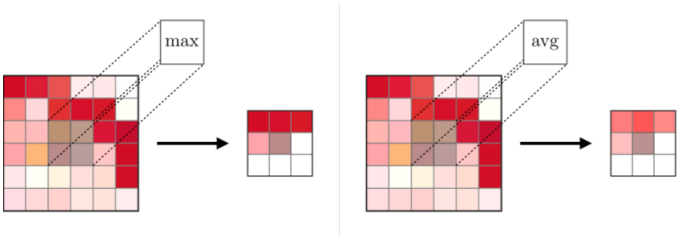


Figure 2.5: Pooling functions max and average. Image taken from [Amidi, 2019]

- **The fully connected (FC) layer:** operates on a flattened input where each input is connected to all neurons. If present, FC layers are usually found towards the end of CNN architectures and can be used to optimize objectives such as class scores.

2.2.4/ CV TASKS USING DEEP LEARNING

In the following, we present swiftly several CV tasks (e.g. image classification, object detection, semantic segmentation, image restoration) that benefited from the success and advancements of different CNN architectures.

2.2.4.1/ IMAGE CLASSIFICATION

Image classification is the process of categorizing an image into a specific group/class. *For instance, a classification process can categorize the image presented in Figure 2.6.(a) as a person, a motorcycle, or a car.* Several DL models such as VGG [Simonyan et al., 2014], ResNet [He et al., 2016a], ResNext [Xie et al., 2017] and

DenseNet [Huang et al., 2017] have been proposed to solve this task by training the network over pairs of image/class annotations. These classifier networks are also employed as feature extractors for neural networks designed to solve other CV tasks (such as object detection or semantic segmentation). Several public datasets such as ImageNet [Russakovsky et al., 2015a], MNIST [LeCun, 1998] and FaceScrub [Ng et al., 2014] provide thousands of image/annotation pairs. The reader can refer to [Khan et al., 2020] for a detailed review regarding image classification with DL.

2.2.4.2/ OBJECT DETECTION

Object detection is the process of detecting the objects' locations in an image (object localization) and classifying them accordingly (object recognition/classification). *For instance, in Figure 2.6.(b), three objects are detected and classified, a person, a car and a motorcycle*. Several DL models such as Yolo [Redmon et al., 2016], SSD [Liu et al., 2016], FasterRcnn [Ren et al., 2015] have been proposed to solve this task via an end-to-end process by training the network via pairs of images/bounding-box annotations. Several public datasets such as MS-COCO [Lin et al., 2014] and Pascal VOC [Everingham et al., 2015] provide thousands of image/bounding-box annotation pairs. The reader can refer to [Jiao et al., 2019] for a detailed review regarding object detection with DL.

2.2.4.3/ SEMANTIC SEGMENTATION

Semantic Segmentation is the process of classifying each pixel in an image. In other words, it creates a pixel-wise mask for the pixels that share similar semantic information in an image. In contrast to object detection, semantic segmentation captures the object's shape, not only its location (c.f. Figure 2.6.(c)). Several DL models such as DeepLab [Zhao et al., 2017] have been proposed to solve this task in an end-to-end manner by training the networks via pairs of image/pixel-level annotations. Several public datasets such as ADE20k [Zhou et al., 2017] and Cityscapes [Cordts et al., 2016] provide thousands of image/pixel-level annotation pairs. The reader can refer to [Minaee et al., 2020] for a detailed review regarding semantic segmentation with DL.

2.2.4.4/ IMAGE RESTORATION

Image restoration is the process of recovering high quality clear images from their degraded counterparts. The degradation can be caused by down-sampling (i.e., pixelating), blurring [Hill et al., 2016b], inpainting (i.e., masking), etc. In this work, we describe briefly the following three image restoration tasks that are most relevant to our study.

(a) (b) (c)

Figure 2.6: (a) Original Image, (b) Object detection , (c) Semantic Segmentation

- **Image super resolution** and more specifically single image super resolution (SISR) is the process of generating a single high-resolution (HR) image from a single low-resolution (LR) image (e.g., downsampled image). Like the other CV applications, several deep neural networks including SRResNet [Ledig et al., 2017], SRGAN [Garcia, 2016] or EDSR [Lim et al., 2017] have been proposed to solve this problem via an end-to-end process by using pairs of low-resolution/high-resolution images. The reader can refer to [Yang et al., 2019b] for a detailed review regarding image super resolution with DL.

- **Image deblurring** is the process of restoring a sharp image from a single image blurred via a blurring kernel (e.g. Gaussian kernel, motion blur...). Several deep neural networks including SRResNet [Ledig et al., 2017] and DeblurGANv2 [Kupyn et al., 2019, Shen et al., 2018a] have been proposed to solve this problem via an end-to-end process by training the networks with pairs of blurred/clear images. Several public datasets such as GoPro [Nah et al., 2017] and RealBlur [Rim et al., 2020] provide thousands of clear/blurred image pairs. The reader can refer to [Koh et al., 2021] for a detailed review regarding image deblurring with DL.

- **Image inpainting** is the process of removing any type of distortion including text, blocks, scratches, or any type of masks by synthesizing the hidden/missing parts of the image. Several deep neural networks have been proposed in [Yeh et al., 2017] to solve this problem via an end-to-end process. The reader can refer to [Pushpalwar et al., 2016] for a detailed review regarding image inpainting.

In this section, we showed how feature extraction is achieved in the context of images. In addition, we presented the basic concepts of artificial neural networks and of convolution neural networks. Last but not least, we presented several CV tasks that we employed throughout our report using DL models.

(a) (b)

Figure 2.7: (a) Image with missing region, (b) Inpainted Image. Image taken from [Pathak et al., 2016].

2.3/ PRIVACY PRESERVATION FOR IMAGES DATASET

After presenting briefly the basic concepts of a neural network and the different CV-based tasks, we discuss in the following section privacy concepts in the context of images that are relevant to Chapters 3, 4 and 5.

2.3.1/ IMAGES DATASET

Processing images without taking into consideration the anonymity of the target identity/individual may lead to privacy breaches because visual features can reveal identifying, quasi-identifying and sometimes sensitive information about her/him.

For instance, processing (e.g. sharing) images of an individual's face (c.f. Figure 2.8.(a)) reveals vast amount of information about her/him such as:

- **Identifying information:** facial features can be used to uniquely identify an individual in a dataset of images.
- **Quasi-identifying information:** facial features can be used as well to recognize quasi-identifying attributes[4] of an individual *such as her/his age, gender, eye color...*.
- **Sensitive information:** facial features can be used to extract sensitive information[5] about an individual. *For instance considering a face images dataset released by a hospital, identifying if a patient has a certain type of skin disease would reveal sensitive information about her/him.*

[4] A quasi-identifying attribute is an attribute that can narrow down the search for an identity when linked to external data sources.

[5] A sensitive attribute reveals critical and sensitive information about a certain individual and must not be directly linked to individuals' identifying values.

Therefore, digital images reveal private information if not properly protected.

Figure 2.8: Obfuscation techniques left to right , (a) Original plain image, (b) pixelated image (4x), (c) Gaussian Blurred Image ($\sigma = 5$), (d) Motion blurred and (e) masking by adding random black pixels. Image taken from [Tekli et al., 2019]

2.3.2/ OBFUSCATION TECHNIQUES

Storing and processing images without guaranteeing the anonymity of the target individuals is a breach of the 'data minimization' principle as discussed in Section 1.4.1. To preserve the individual's anonymity, numerous obfuscation techniques have been proposed in the literature to hide/protect features in images such as (i) the traditional techniques (e.g. pixelating, blurring, masking), (ii) the k-same methods [Newton et al., 2005a] or (iii) the GAN-based inpainting approaches [Hao et al., 2019]. Nowadays, the majority of social media platforms, news agencies and publicly available research datasets still use the traditional techniques such as pixelating or blurring: *for instance, Google Maps [Frome et al., 2009] as well as the large-scale dataset nuScenes [Caesar et al., 2020] published in 2019 for autonomous driving still employ blurring kernels to obfuscate individuals' faces/homes or vehicle plates.* Therefore, we focus in this study on the following three obfuscation techniques: pixelating, blurring and masking.

- **Pixelating** (a.k.a. mosaicking) is widely adopted as an obfuscation technique. The identifying/sensitive information to be obfuscated is divided into a square grid, a.k.a. "a pixel box". Each pixel box will have one color after averaging the values of the grouped pixels in it [Hill et al., 2016b]. The size of the pixel box can be modified depending on the needed level of privacy. The larger the box, the more pixels will be averaged together, the higher the level of privacy. As stated in [McPherson et al., 2016], although the size of the image stays the same, pixelating can be thought of as reducing the obfuscated section's resolution. For instance, downscaling an image by a factor of 4 is equivalent to applying a pixel box of size 4x4.(c.f. Figure 2.8.b).

- **Blurring** is also a degradation technique utilized in image processing. It can be generated by a Gaussian kernel or via a camera motion effect, a.k.a. motion blur. A Gaussian like blur kernel is used extensively as an obfuscation technique [Hill et al., 2016a]. It removes details from an image by applying a Gaussian kernel. The blurriness level is controlled by the standard deviation σ. A motion blur alters the details of an image by generating the effect of a synthetic camera motion blur [Boracchi et al., 2012]. The level of blurriness is affected by the length and the angle of the synthesized motion (c.f. Figure 2.8(c-d)).

- **Masking** removes details from an image by replacing the original pixels by black pixels. The masking technique can have multiple derivatives depending mainly on the color intensity and location of the altered pixels. For instance, if an individual's face is considered sensitive, pixels can be modified around the eyes and nose or at random points of the face. The level of privacy depends on the amount, location and color intensity of the modified pixels (c.f. Figure 2.8.e).

In Chapter 3, we designed and implemented an anonymization tool (i) that localizes via DL-based approaches (e.g., object detection and semantic segmentation presented) identifying/sensitive features in images/videos and (ii) obfuscates them via pixelating, blurring or masking.

Numerous studies focus on the validity of these techniques from different perspectives such as privacy, intelligibility, viewer's perception, etc. Please refer to the related works section in Chapter 4 for more details.

2.3.3/ Recognition and Restoration-based attacks

Obfuscation is done by altering/removing features from the images to hide identifying information while, at the same time, retaining some visual features to keep the image suitable for processing. However, these visual features can be used to identify/reconstruct the obfuscated private information via different attacks that can be classified as *recognition*-based [McPherson et al., 2016, Newton et al., 2005a, Lander et al., 2001] and *restoration*-based attacks [Ruchaud et al., 2016, Abramian et al., 2019].

- ***Restoration*-based attacks** de-anonymize privacy-protected images by trying to restore/reconstruct the plain original features of the obfuscated information [Ruchaud et al., 2016, Keys, 1981]. *For instance, the authors in [Lander et al., 2001] cancels the impact of pixelating, blurring and masking with regard to face recognition algorithms by applying ad-hoc traditional image reconstruction techniques (e.g., bicubic interpolation [Keys, 1981].*

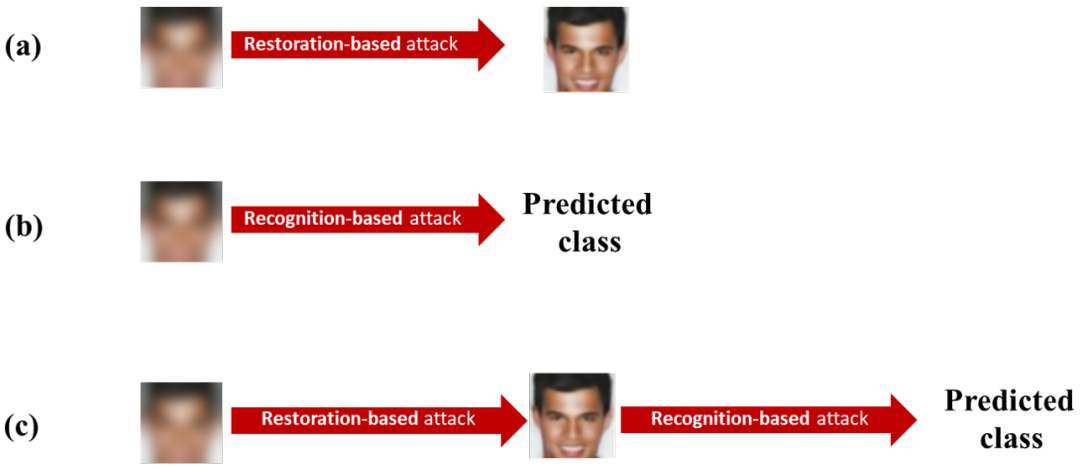


Figure 2.9: Attacking scenarios: (a) *Restoration*-based attack, (b) *Recognition*-based attack, (c) *Restoration&Recognition*-based attack

- ***Recognition*-based attacks** breach the images privacy and anonymity by training a classifier to perform recognition tasks on obfuscated information. *For instance, the authors in [McPherson et al., 2016] demonstrate that obfuscated faces, objects and digits can be recognized by deep neural networks trained via a supervised manner on obfuscated/clear images.*

- ***Restoration & Recognition*-based (R&R) attacks** perform a 2-steps attack against an obfuscation technique by (1) restoring the hidden features of an obfuscated image and (2) trying to associate the restored features with an identifying attribute, i.e. classifying the restored image.

As mentioned in Section 2.2, DL outperforms traditional learning-based approaches with regard to the different CV tasks [Russakovsky et al., 2015b, Yang et al., 2019b]. Hence, from a privacy perspective, these DL-based techniques are highly nominated as strong *recognition*-based and *restoration*-based attacks [McPherson et al., 2016, Hao et al., 2020].

The rise of such DL-assisted attacks complicates the process of choosing the most robust obfuscation technique when anonymizing an images dataset. Therefore, we propose in Chapter 4 a quantitative recommendation framework that evaluates the robustness of several obfuscation techniques against adversaries performing DL-assisted attacks. In addition, we study how the adversary's capabilities (i.e., DL-assisted attacks) scale along with her/his knowledge about the target dataset. We also demonstrate in Chapter 5 that adversaries lacking knowledge and data, can leverage their attacking capabilities and increase the privacy breaches by collaborating with other adversaries via FL.

2.4/ PRIVACY PRESERVATION FOR STRUCTURED RELATIONAL DATASETS

After discussing privacy preservation for images dataset, in the following section we present several privacy preserving mechanisms and privacy models in the context of structured relational datasets. The following section is relevant to Chapter 6.

2.4.1/ STRUCTURED RELATIONAL DATASET

As mentioned in Chapter 1, a structured relational (tabular) dataset consists of a table with different attributes and tuples. Each table can be shared/released as macrodata or microdata [Bayardo et al., 2005]. In a macrodata release, each tuple represents an aggregated statistic computed over a sample population. Whereas in a microdata release, each tuple represents an entity/individual-related information. In this thesis and more specifically in Chapter 6, we focus on microdata releases, hence a tuple is a finite ordered list of values that corresponds to certain attributes. The attributes can be categorized as follows:

- **Identifying attribute (A^{id})**: is unique and linked to a single individual in a given dataset.
- **Quasi-identifying attribute (A^{QI})**: can narrow down the search for an identity when linked to external data sources.
- **Sensitive attribute(A^{sv})**: reveals critical and sensitive information about a certain individual and must not be directly linked to individuals' identifying values in data sharing, publishing or releasing scenarios.

We consider the following example scenario to elaborate on the privacy concepts that are relevant to Chapter 6.

Example Scenario Let us consider in Figure 2.10.(a) a microdata table released by a hospital with regard to its patients' drug use. The table is composed of four attributes: (i) "User_ID" is considered the identifying attribute (A^{id}), (ii) "Age" and "Gender" are considered the quasi-identifying attributes (A^{QI}) and (iii) "Drug Name" is considered the sensitive attribute (A^{sv}).

2.4.2/ PRIVACY PRESERVING MECHANISMS

Sharing a microdata table in its original form disclose the identities of the tuples' owners as well other sensitive information. Throughout the years, several privacy preserving

mechanisms/techniques have been proposed to pre-process the microdata table prior to release and prevent such disclosures. These mechanisms can be classified as (i) perturbative and (ii) non-perturbative mechanisms/techniques [Mehmood et al., 2016]. On the one hand, a perturbative mechanism adds synthetic tuples so that the statistical information computed from the released data does not differ significantly from that of the original data [Dwork, 2008]. On the other hand, a non-perturbative mechanism sanitizes/modifies the original identifiable and quasi-identifiable values within the dataset [Samarati et al., 1998]. In the following, we elaborate on the relevant ones to our study:

- **Suppression** (non-perturbative): obfuscates the original value of an identifiable or quasi-identifying attribute with a special character/value (e.g.*). *For example, replacing the (A^{QI}) "Gender" values for patients U1 and U2 in Figure 2.10.(b)) by *.*
- **Generalization** (non-perturbative): transforms the original values into less specific but semantically consistent values. *For example, the value 20 of the (A^{QI}) "Age" is generalized with an interval [20-28] in Figure 2.10.(b)).*
- **Bucketization** (non-perturbative): splits the original table into two by separating the QI attributes and the sensitive attributes and adds noise to the level of association between the sensitive and identifying attributes (c.f. Figure 2.11.(b)).

2.4.3/ PRIVACY THREAT DISCLOSURES

Although these privacy preserving mechanisms increase the level of privacy, time has shown that they fail to protect individual's anonymity. As stated in [Majeed et al., 2020], two major factors could lead to several privacy disclosures and jeopardize the individuals' anonymity: (i) the knowledge of an adversary about the target dataset and (ii) flaws within the privacy mechanism applied. Below, we list two privacy disclosures that are most relevant to our study:

- **Identity disclosure**: arises when an adversary links an individual/identity to a particular tuple in the released dataset. Generally, an adversary can cause such disclosure when equipped with certain knowledge (e.g., knowing that a certain individual has a record within the dataset) and using additional data via external sources to re-identify the individual. *For instance, the authors in [Sweeney, 2000] proved that suppressing the identifiers within a microdata dataset is not enough to preserve the data owners' anonymity.*
- **Attribute disclosure**: arises when an adversary links an individual's identifying or quasi-identifying values to his/her corresponding sensitive information. This disclosure usually occurs against an imbalanced dataset, i.e., due to a flaw of the privacy

preserving mechanism (when the dataset lacks heterogeneity in terms of sensitive values, i.e. homogeneity attack [Machanavajjhala et al., 2007]).

2.4.4/ PRIVACY MODELS

Multiple privacy models formalized the privacy preserving mechanisms discussed in Section 2.4.2 to avoid identity and attribute disclosures. Below, we elaborate on the most relevant ones to our study:

- **k-anonymity** [Sweeney, 2002]: acts against identity disclosure by dividing the dataset into groups of k tuples where all the k tuples share the same QI-value. In other words, each combination of quasi-identifier attribute values is shared by a group of at least k tuples, i.e. QI-group. *For instance, in table 2.10.(b) the adversary cannot link an external tuple to the released dataset based on the quasi-identifying attributes "age" and "gender" with a probability greater than* $1/k$.

However, as shown in [Machanavajjhala et al., 2007], k-anonymity might suffer from attribute disclosure even with large k values due to homogeneity attacks. Therefore l-diversity was introduced.

- **l-diversity** [Machanavajjhala et al., 2006]: acts against attribute disclosure by diversifying the values of the sensitive attribute within a single QI-group. Basically, l-diversity requires the presence of at least l well-represented (different) values for the sensitive attributes in a QI-Group.

A dataset is made k-anonymous and l-diverse by generalizing the original values as seen in Figure 2.10.(b). The released table in Figure 2.10.(b) is a *2*-anonymous and a *2*-diverse version of the original microdata table with 4 QI-groups.

Additional privacy preserving models were proposed and studied throughout the years to address the limitations of both k-anonymity and l-diversity such as t-closeness [Li et al., 2007], differential privacy [Dwork et al., 2016], etc. The reader can refer to [Awad, 2020, Majeed et al., 2020] for a detailed review of the different privacy preserving models.

User_ID	Age	Gender	Drug Name
U1	22	M	Mild Exfoliation
U2	28	F	Retinoic acid
U3	20	M	Azelaic acid
U2	28	F	Retinoic acid
U5	27	M	Cytarabine
U3	20	M	Adapalene
U4	21	F	Cytarabine
U4	21	F	Azelaic acid

(a) Original table

User_ID	Age	Gender	Drug Name
U1	[22,28]	*	Mild Exfoliation
U2	[22,28]	*	Retinoic acid
U3	[20,28]	*	Azelaic acid
U2	[20,28]	*	Retinoic acid
U5	[20,27]	M	Cytarabine
U3	[20,27]	M	Adapalene
U4	21	F	Cytarabine
U4	21	F	Azelaic acid

(b) 2-anonymous and 2-diverse table

Figure 2.10: Applying k-anonymity and l-diversity to a microdata release

2.4.5/ GENERALIZATION VS BUCKETIZATION

As mentioned in Chapter 1, data utility plays a major role when it comes to data anonymization. Generalizing the original values to satisfy the privacy constraints (i.e. k-anonymity [Sweeney, 2002] or l-diversity [Machanavajjhala et al., 2006]) often leads to considerable loss of information (c.f. Figure 2.10.(b)) especially with high-dimensional quasi-identifying attributes [LeFevre et al., 2005]. Therefore, applying bucketization-based mechanisms such as anatomy [Xiao et al., 2006a] might lead to less information loss and better utility. Anatomy groups k tuples into l-diverse QI-groups and produces two tables: a QI table and a sensitive table connected via a group ID as seen in Figure 2.11.(b)). The QI-group size and grouping of tuples into QI-groups via ensures that privacy constraints (such as k-anonymity [Sweeney, 2002] or l-diversity [Machanavajjhala et al., 2006]) are satisfied.

2.4.6/ CORRELATION PROBLEM IN TRANSACTIONAL DATASETS

Although bucketization-based techniques [Xiao et al., 2006b, Li et al., 2012, Ciriani et al., 2010, Terrovitis et al., 2012] might lead to better data utility compared to generalization-based techniques [Campan et al., 2011, He et al., 2009, Anjum et al., 2017] however they both assume a trade-off between data privacy and utility. It is a trade-off that is highly required to keep the dataset suitable for analysis while preserving the individuals' anonymity. However, it keeps anonymization vulnerable and unable to cope with different sort of attacks [Wong et al., 2011a, Cormode et al., 2010, Kifer, 2009a, al Bouna et al., 2015b]. It is indeed difficult to provide a completely anonymous dataset without losing utility. There are many reasons for this to happen, notably, is the ability to presume knowledge of the adversary's prior belief and her/his ability to gain insights after looking at the anonymized dataset. For instance a transactional dataset[6] may expose significant correlations between identifying and sensitive values. An adversary can use her/his knowledge

[6] In a transactional dataset, an individual might have multiple tuples.

of such correlations [Wong et al., 2011b, Kifer, 2009a], or use these correlations as foreground knowledge [Li et al., 2008] to breach individuals' privacy.

As stated in [al Bouna et al., 2013], inter and intra QI-group correlations may arise when applying Anatomy [Xiao et al., 2006b] to a transactional dataset:

- **Inter-group correlation**: occurs when an adversary exposes a correlation between identifying and sensitive values by observing the different released QI-groups. *For instance in Figure 2.11.(b), the individual "U2" appears in all the QI-groups with the "Retonic Acid" drug, hence it is likely that "U2" is taking that drug. The adversary might have certain facts about drug use as background knowledge (e.g. Retonic Acid is a maintenance drug taken over a long period of time) or might learn some drug use facts from the anonymized dataset and use it as foreground knowledge.*
- **Intra-group correlation**: occurs when an adversary exposes a correlation within a single QI-group where the number of transactions for a single individual results is an inherent violation of l-diversity. *For instance while observing the 4th QI-group in Figure 2.11.(b), the adversary knows that the individual "U4" is taking both drugs "Cytarbine" and "Azetaic Acid" because all tuples in the QI-group belong to her/him.*

User_ID	Age	Gender	Drug Name
U1	22	M	Mild Exfoliation
U2	28	F	Retinoic acid
U3	20	M	Azelaic acid
U2	28	F	Retinoic acid
U5	27	M	Cytarabine
U3	20	M	Adapalene
U4	21	F	Cytarabine
U4	21	F	Azelaic acid

(a) Original table

User_ID	Age	Gender	GID
U1	22	M	1
U2	28	F	1
U3	20	M	2
U2	28	F	2
U5	27	M	3
U3	20	M	3
U4	21	F	4
U4	21	F	4

GID	Drug Name
1	Mild Exfoliation
1	Retinoic acid
2	Azelaic acid
2	Retinoic acid
3	Cytarabine
3	Adapalene
4	Cytarabine
4	Azelaic acid

(b) Anatomized table

User_ID	Age	Gender	GID
U2	28	F	1
U3	20	M	1
U2	28	F	2
U3	20	M	2
U4	21	F	3
U5	27	M	3
U1	22	M	3
*	*	*	3

GID	Drug Name
1	Mild Exfoliation
1	Retinoic acid
2	Azelaic acid
2	Retinoic acid
3	Cytarabine
3	Adapalene
3	Cytarabine
3	Azelaic acid

(c) Safe Grouped table

Figure 2. Bucketization-based techniques

Figure 2.11: Bucketization-based techniques

To cope wih both the intra/inter correlation problem, safe grouping is proposed in [Li et al., 2008, al Bouna et al., 2013] to ensure that the individuals' tuples are grouped in one and only one QI-group that is at the same time l-diverse, respects a minimum diversity for identifying attribute values and all individuals in the same QI-group have an equal number of tuples (c.f. Figure 2.11). *For instance in Figure 2.11.(c), we notice that both "U2" and "U3" appear with "Retonic Acid" throughout the dataset. Hence, we cannot single out "U2" in this case as in Figure 2.11.(b) and potentially link her/him to the drug "Retonic Acid".* (k,l)-diversity [Gong et al., 2017] is another technique that uses generalization to associate k distinct individuals to l-diverse QI-groups. Both techniques were developed to deal with the correlation problem in the context of bulk static datasets. However, these techniques provide no proof of effectiveness in anonymizing a data stream where data must be processed and protected on the fly. Hence, we propose in Chapter 6 a clustering-based approach, entitled (k, l)-clustering, that anonymizes a transactional

data stream on the fly while taking into consideration the inter/intra-group correlations' problem.

2.5/ CONCLUSION

This chapter serves as preliminary to Part III. We mainly presented basic privacy concepts regarding two data types: (i) digital images and (ii) relational structured data. First of all, we presented the basic DL concepts and several tasks for CV (e.g. classification [Krizhevsky et al., 2012], object detection [Jiao et al., 2019], image restoration [Koh et al., 2021, Pushpalwar et al., 2016, Nasrollahi et al., 2014]). Then, we presented several obfuscation techniques applied in the context of images (e.g. blurring, pixelating and masking [Hill et al., 2016b]) and several attacks that try to defeat them (e.g. *restoration*-based and *recognition*-based attacks). These concepts are relevant to Chapters 3, 4 and 5. In Chapter 3, we design and implement an obfuscation tool that localizes via DL-based approaches (e.g. object detection [Jiao et al., 2019] and semantic segmentation [Chen et al., 2018]) sensitive/identifying features in images/videos and obfuscates them. Whereas, in Chapter 4, we propose a quantitative recommendation framework that evaluates the robustness of several obfuscation techniques against adversaries performing DL-assisted attacks. In addition, we study how the adversary's capabilities (i.e., DL-assisted attacks) scale along with her/his knowledge about the target dataset. We also demonstrate in Chapter 5 that adversaries lacking knowledge and data, can leverage their attacking capabilities and increase the privacy breaches by collaborating with other adversaries via FL. Second, we presented the different privacy preserving mechanisms in the context of structured relational datasets (e.g., suppression, generalization, bucketization) [Mehmood et al., 2016, Dwork et al., 2006], the different privacy threat disclosures (e.g., identity and attribute disclosure [Sweeney, 2000, Machanavajjhala et al., 2007]) and some privacy models applied to avoid these privacy disclosures (e.g., k-anonymity [Sweeney, 2002] and l-diversity [Machanavajjhala et al., 2006]). Afterwards, we showed how anatomy [Xiao et al., 2006a] might lead to inter-group and intra-group correlation problems [al Bouna et al., 2013] when applied to transactional datasets and how certain techniques were developed to prevent it in the context of static bulk datasets [Gong et al., 2017, al Bouna et al., 2013]. However, as these techniques were developed for static datasets, they do not cope well with transactional data streams where new tuples are generated at each instance, hence our contribution in Chapter 6.

The reader can also refer to Chapters 3,4,5 and 6 for a deep dive into the related works of each contribution.

III

CONTRIBUTIONS

3

Image Obfuscation Tool at BMW Group

3.1/ Scenario and problem definition

As mentioned in Chapter 1, manufacturing units at BMW Group are increasingly integrating DL models for CV-based applications mainly to assist plant workers with the quality checks throughout the production line. As a result, a huge number of images is being captured, stored and processed by data scientists on a daily basis. However, these images might contain identifying/sensitive features (e.g., worker's face or name tag) which are not relevant to the DL workflow. Therefore, we designed and implemented an obfuscation/anonymization[1] tool that **localizes** and **obfuscates** identifying/sensitive information in images/videos (c.f. Figure 3.1). Several obfuscation/anonymization tools are available today on the market [eyedea Regonition, 2021, AI, 2018, brighter AI, 2019, sightengine, 2020] however not a single one combines the following features: (i) scalable in terms of obfuscation techniques, (ii) agnostic in terms of localization approaches, (iii) modular in terms of identifying/sensitive information, (iv) GDPR compliant, (v) open-source[2] and (vi) compatible with other DL-based tools for CV tasks developed at BMW Group[3]. This work was published as a public GitHub Repository [Tekli et al., 2021], as a press release by BMW Group [Hatzel, 2021] and as a white paper by Intel Cooperation [Intel, 2021].

The remainder of this chapter is organized as follows. In Section 3.2, we present our anonymization tool along with its features. In Section 3.3, we take a look at the available anonymization tools available today on the market and we elaborate on what distinguishes our proposed solution.

[1] Throughout this chapter, we will use the terms obfuscation and anonymization interchangeably.
[2] https://github.com/BMW-InnovationLab/BMW-Anonymization-API
[3] https://github.com/BMW-InnovationLab

3.2/ PROPOSED ANONYMIZATION TOOL

We designed our anonymization tool as a micro-service that receives an image along with a JSON object through which the user specifies: (i) the identifying/sensitive information she/he wishes to obfuscate, (ii) the obfuscation technique, (iii) the obfuscation degree and (iv) the localization method she/he wishes to employ[4]. *For instance in Figure 3.1, the user wishes to* ***localize*** *"persons" appearing in the input image via "semantic segmentation" and* ***obfuscate*** *them via the "full masking technique" with "degree 1".*

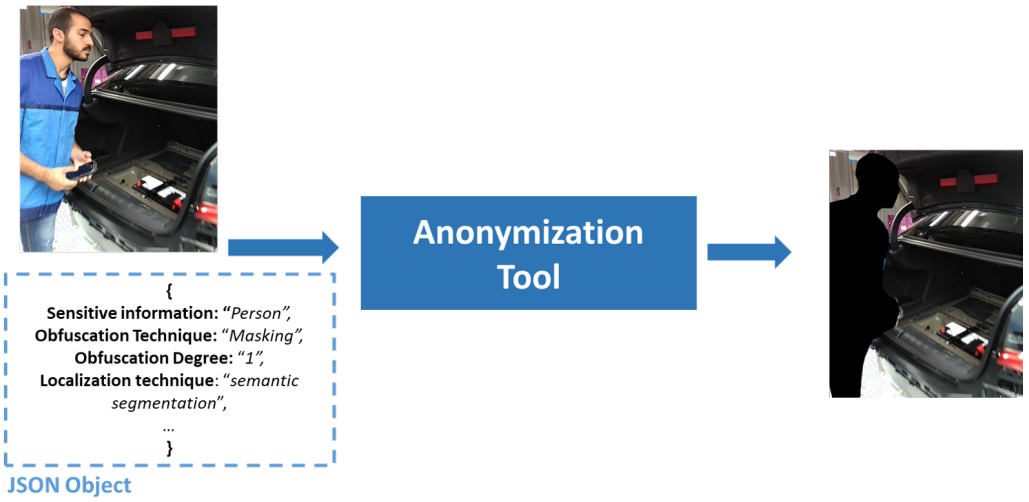


Figure 3.1: Proposed Anonymization Tool

The anonymization tool parses the received data and triggers a 2-layered iterative workflow in order to obfuscate the identifying/sensitive information. The anonymization tool is composed of 2 units: (1) the localization unit and the (2) obfuscation unit (c.f. Figure 3.2).

[4] Additional information are specified in the JSON object however we indicate the most relevant ones.

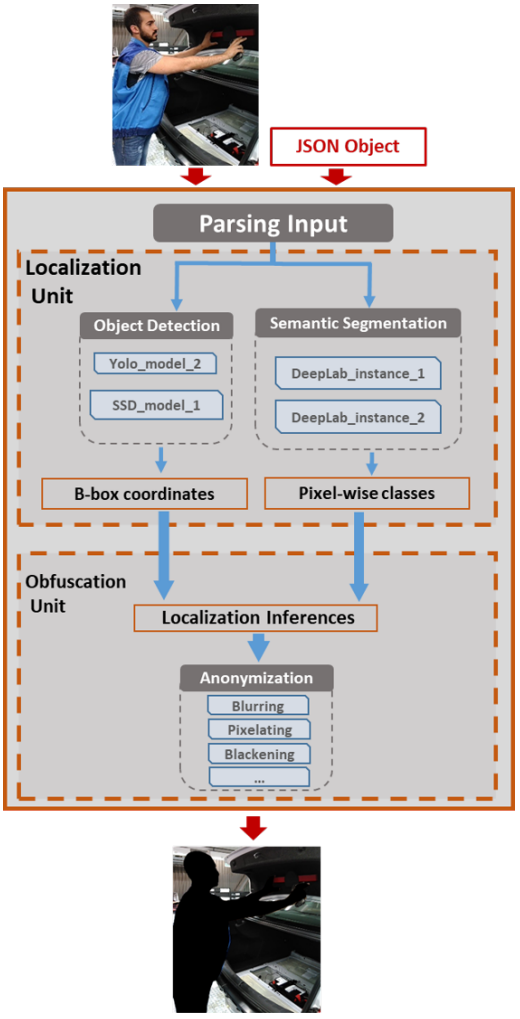


Figure 3.2: A 2-layered architecture

- **Localization Unit** : contains instances of DL models trained to localize certain objects in an image. These DL models can either perform object detection (e.g. Yolo [Redmon et al., 2016], SSD [Liu et al., 2016] or FasterRcnn [Ren et al., 2015]) or semantic segmentation (e.g. DeepLab [Chen et al., 2018]). Based on the JSON object and more specifically the *identifying/sensitive information* and the *localization technique* provided by the user, the input image is mapped to the corresponding DL model[5]. After inferring over the input image, the localization unit forwards the image along with the inferences[6] to the obfuscation unit (c.f. Figure 3.3).

- **Obfuscation Unit** : anonymizes the localized portion (e.g. bounding boxes or per-pixel classes) via the *obfuscation technique* and the *degree* specified in the input

[5] *For instance, let us consider a DL model "Deeplab_instance_1" that supports the semantic segmentation of a "person". This DL-model is present in the localization unit. If the user specifies in the input JSON Object "person" as sensitive information and "semantic segmentation" as localization technique, then the input image will be mapped to DL model "DeepLab_instance_1".*

[6] If the DL-model performs object detection, then it infers bounding box along with the classes. Whereas if the DL-model performs semantic segmentation, it infers per-pixel classes, i.e. segments.

JSON object (c.f. Figure 3.3).

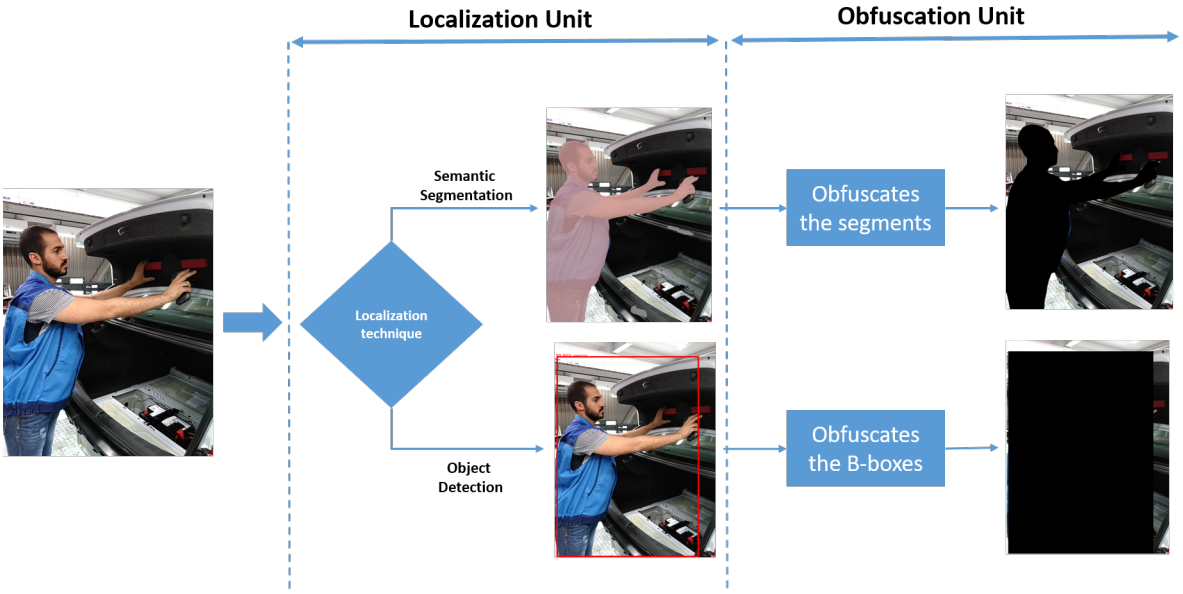


Figure 3.3: A 2-steps workflow

3.2.1/ ANONYMIZATION TOOL'S FEATURES

As mentioned before, we designed our anonymization tool with the following characteristics:

1. **Scalable in terms of obfuscation techniques and degrees**: as mentioned in Chapter 2, different obfuscation techniques are proposed in the literature such as blurring, pixelating [Hill et al., 2016b], masking, face swapping [Hao et al., 2019], etc. Numerous studies investigate the validity of each obfuscation technique with regard to different sensitive information and evaluation metrics [Tekli et al., 2019, Hao et al., 2020] however to date there is no clear nor unified choice with regard to which obfuscation to employ in each scenario. Therefore, we support in our tool not only one but multiple obfuscation techniques. In addition, we offer the user full flexibility with regard to the privacy-utility trade-off where she/he can specify the degree of anonymization ranged between 0 and 1, 1 being the highest level of anonymization (c.f. Figure 3.4).

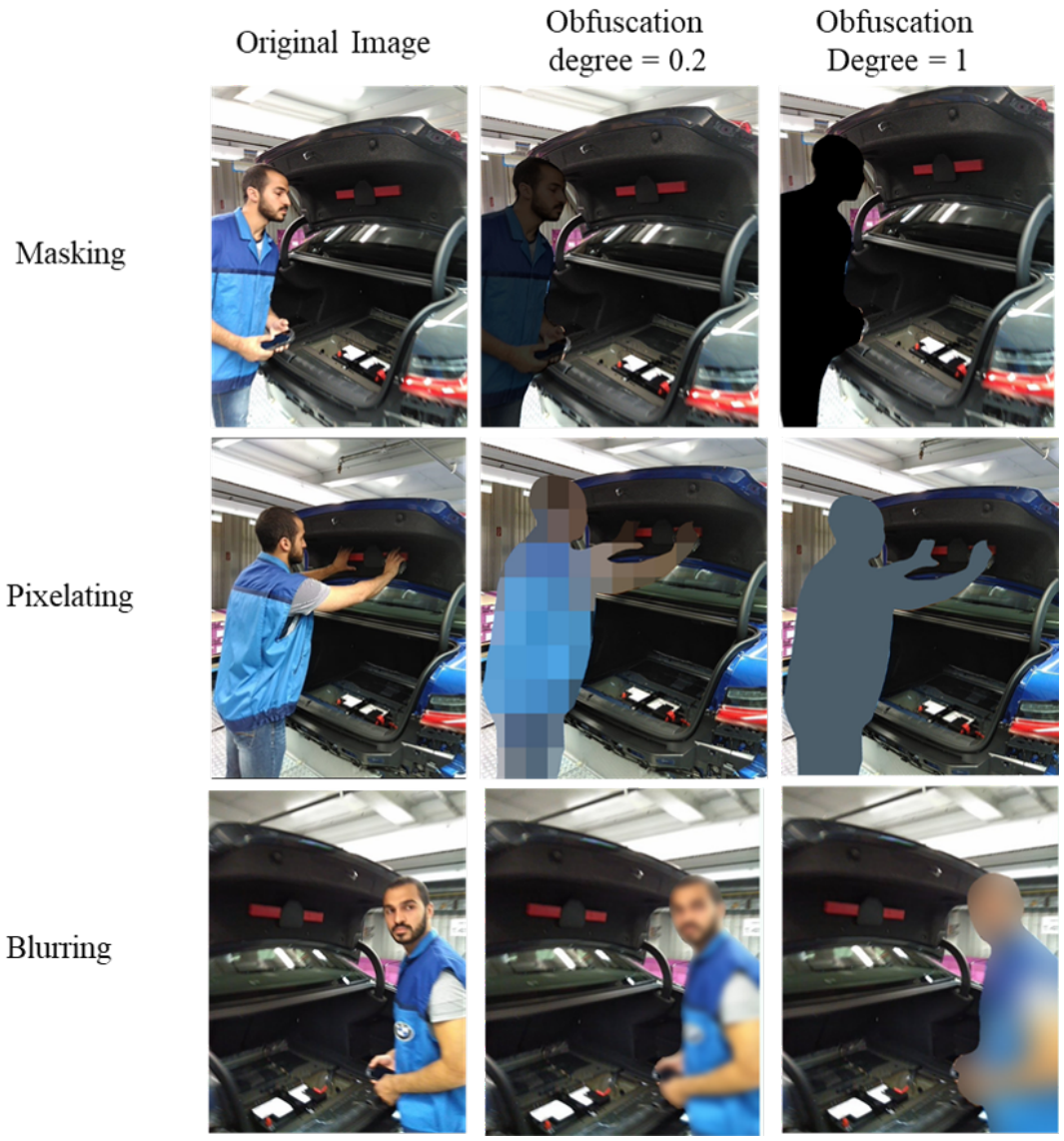


Figure 3.4: Scalibility in terms of obfuscation techniques and degrees.

2. **Agnostic in terms of localization approaches**: as mentioned in Chapter 2, we can employ different DL-based approaches to localize an object in an image. Some DL models are designed to detect objects via bounding boxes, i.e. object detection whereas others are designed to classify each pixel connected to the target object, i.e. semantic segmentation. The proposed anonymization tool supports both object detection and semantic segmentation[7](c.f. Figure 3.3). A scenario where we might/can use the two localization techniques simultaneously is shown in Figure 3.5.

[7]Other localization techniques, such as instance segmentation, can be supported but for our use cases, object detection and semantic segmentation were sufficient.

(a) Original Image (b) Obfuscated Image

Figure 3.5: Obfuscating the license plates and the individuals on the street. Image taken from the cityscape dataset [Cordts et al., 2016]

3. **Modular in terms of identifying/sensitive information:** as mentioned in Section 3.2, the localization unit contains DL models for localizing different objects via detection and segmentation. Therefore, we can support the anonymization of identifying/sensitive information as long as we train a DL model to localize it and add this DL model as an instance in the localization unit (c.f. Figure 3.2). At the BMW Group manufacturing units, identifying/sensitive information are sometimes specific to the plant environment. *For instance, closets where workers store their personal belongings sometimes appear in images taken at the manufacturing units and are considered highly sensitive information. In that case, we need to train a DL model (either for object detection or semantic segmentation) to localize these specific closets and anonymize them accordingly.*

4. **GDPR compliant:** as mentioned in Chapter 1, privacy regulations [European Parliament, 2018b] clearly state that the individual's privacy should be anonymized with irreversible effect. Based on Chapters 4, 5 and other studies [Hao et al., 2020], we notice that blurring, pixelating and even random masking are not bullet proof against restoration/re-identification attempts. Therefore, we always recommend to employ full black masking with degree 1 in order to avoid any restoration/re-identification attempts (c.f. Figure 3.4). However, if the user wishes to employ other obfuscation techniques with lower degrees, then she/he should study the robustness of the obfuscation via the recommendation framework proposed in Chapter 4 before employing the obfuscation technique in the anonymization tool.

5. **Open-Source:** Unlike the solutions available today on the market, our anonymization tool is accessible as a public repository on GitHub for the researcher/developer communities [Tekli et al., 2021].

6. **BMW Compatible:** At BMW Group, we developed end-to-end DL-based solutions for CV applications (i.e., Image classification, Object detection , semantic/instance segmentation) for training/inference purposes. These solutions are published on

GitHub[8]. The proposed anonymization tool is fully compatible with these training/inference solutions.

3.3/ RELATED WORKS

Multiple anonymization tools and solutions are available today on the market however none of them combine the six features we listed in Section 3.2.1. For instance, Eyedea Recognition [eyedea Regonition, 2021] offers an anonymization solution that only detects faces along with license plates and applies blurring filters on top. Also, the developers at UnderstandAI offer a similar anonymization tool and they published a demo version of their code on github [AI, 2018] where they provide the user the option to use models trained via the tensorflow Object detection API [Huang et al., 2021]. In addition, Celantur is another company that offers a tool that supports both segmentation and detection for anonymization [Celantur,]. All of the above solutions do not offer scalability in terms of obfuscation techniques and degrees (e.g. only support blurring) nor modularity in terms of sensitive information (they are limited to 4 identifying/sensitive information, e.g. faces, bodies, license plates and vehicles). In addition, they are not open-source.

On another note, the company docbyte offers another automated anonymization tool as a micro-service that obfuscates sensitive information in documents such as texts and faces on identity cards [docbyte,]. They offer different obfuscation techniques (blurring and masking) however they are not scalable in terms of identifying/sensitive information nor agnostic in terms of localization techniques (e.g. they only employ object detection). Also, they are not open-source.

In addition, Vaisala's anonymization service is capable of detecting and anonymizing vehicles and pedestrians in images and videos [vaisala, 2018]. They employ semantic segmentation to localize the sensitive information. They claim that they are always improving their DL models by training on newly generated data however similar to the above solutions they only consider anonymizing specific features such as vehicles and individuals and they are not open-source. Also, they are not agnostic in terms of localization techniques, nor scalable in terms of obfuscations techniques and degree. Similar to Vaisala's solution, Sightengine [sightengine, 2020] also offers an anonymization tool as a micro-service that hides faces and texts via blurring technique. Last but not least, BrighterAI offers a proprietary anonymization tool that employs DL models to replace individuals faces and license plate numbers with synthetically generated features [brighter AI, 2019].

[8]https://github.com/BMW-InnovationLab

3.4/ CONCLUSION

In this chapter, we designed and implemented an obfuscation tool that localizes and obfuscates (i.e. hides) sensitive/identifying features in images/videos in order to preserve the individuals' anonymity. Several tools are available today on the market however what differentiates ours is that it is at the same time (i) agnostic in terms of localization techniques, (ii) modular in terms of sensitive information, (iii) scalable in terms of obfuscation techniques and (iv) compatible with other DL-based tools for CV tasks such as object detection and semantic segmentation developed at BMW Group[9]. This work was published as a public GitHub Repository [Tekli et al., 2021], as a press release by BMW Group [Hatzel, 2021] and as a white paper by Intel Cooperation [Intel, 2021] as seen in Figure 3.6.

(a) BMW Group press release

(a) Intel Cooperation white Paper

Figure 3.6: Press release regarding the anonymization tool

[9]https://github.com/BMW-InnovationLab

4

A Framework for Evaluating Image Obfuscation under Deep Learning-Assisted Privacy Attacks

4.1/ Scenario and problem definition

As mentioned in Chapter 1, privacy regulations compel data-driven companies to guarantee a level of anonymization[1] that requires "irreversibility preventing identification of the data subject", taking into account all the means "reasonably likely to be used" for identification. This statement implores us to ask ourselves the following question when anonymizing a dataset of images via the tool proposed in Chapter 3: "*What is the most robust image obfuscation technique that guarantees individuals' anonymity against an adversary performing any sort of attack?*"

As defined in [Do et al., 2018] within the field of security, an adversary refers to an attacker, often with malicious intents, that undertakes an attack on a secure system to prevent or disrupt its proper operation. The authors in [Do et al., 2018] presented the adversary as a three-component model having a **goal**, **assumption** (e.g. knowledge) and **capabilities**.

Example Scenario Let us consider an adversary who has access to a dataset of obfuscated faces belonging to certain individuals and her/his goal is to recover their identities. On the one hand, the adversary is capable of performing a *recognition*-based, a *restoration*-based or an *R&R*-based attack in order to extract the needed information from the anonymized faces (c.f. Section 2.3.3). On the other hand, undertaking these attacks

[1] Throughout this chapter, we will use the terms obfuscation and anonymization interchangeably.

depends heavily on the adversary's knowledge with regard to the anonymized dataset, more specifically her/his background knowledge. For instance, the adversary should only be aware of the obfuscation technique employed in the anonymized dataset when performing a *restoration*-based attack. Whereas, she/he is capable of performing an identity *recognition*-based or an *R&R*-based attack only when equipped with knowledge related to the identities $present$[2] in the target anonymized dataset.

Contributions Therefore, in order to consider different scenarios w.r.t. the adversary's attacks and provide a thorough evaluation of the obfuscation techniques we:

1. Proposed a quantitative recommendation framework that evaluates the robustness of image obfuscation techniques and recommends the most resilient obfuscation against DL-assisted attacks.
2. Embedded and adapted the three-component adversary model presented in [Do et al., 2018] to our application domain, facial image obfuscation.
3. Defined different threat levels, with regard to the adversary's background knowledge, where she/he can perform *restoration*-based, *recognition*-based and *R&R*-based attacks.

The remainder of this chapter is organized as follows. In Section 4.2, we present the recommendation framework, formalize the adversary and consider three threat levels. Section 4.3 evaluates different faces obfuscation techniques via the proposed framework and study the effect of the background knowledge on the adversary's capabilities. In Section 4.4, we present how our recommendation framework can be extended to other identifying information and scaled to include different adversaries, DL-assisted attacks and evaluation metrics. In Section 4.5, we investigate works related to privacy attacks in the context of images and to evaluation frameworks. This chapter was published partially in the 17th International Conference on Privacy, Security and Trust (PST) in 2019 [Tekli et al., 2019] and is currently under peer review in a scientific journal [Tekli et al., NDb]

4.2/ PROPOSED FRAMEWORK

In this section, we introduce the recommendation framework by (i) presenting a 4-layered iterative workflow inspired by the KDD process [Fayyad et al., 1996], (ii) showcasing the framework's detailed structure when applied to a facial images dataset and (iii) defining the 3-components adversary model with three threat levels and different attacking capabilities. The framework attacks obfuscation techniques by restoring/recognizing hidden

[2]That possess obfuscated face images

facial features, evaluates the reconstruction/recognition and suggests the most resilient obfuscation. The framework is mainly composed of four units: (a) a data preparation unit, (b) an adversary unit, (c) an evaluation unit and (d) an interpretation unit (c.f. Figure 4.1).

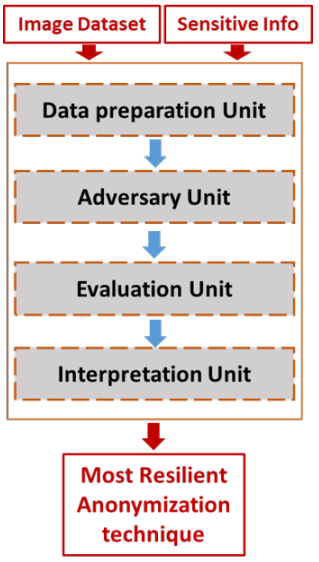


Figure 4.1: The generic recommendation framework

4.2.1/ DATA PREPARATION UNIT

The data preparation unit takes as inputs an image dataset along with the identifying/sensitive information. It is divided into two modules: (a) *detector* and (b) *obfuscator* (c.f. Figure 4.2). As its name indicates, the *detector* localizes and detects the identifying information in the image, crops it and sends it to the *obfuscator*. As stated before, we consider in this study the faces as identifying features. Hence, the *detector* employs the OpenFace toolbox [Amos et al., 2016] to detect faces in an image, crop and forward it to the *obfuscator*. In this study, the *obfuscator* anonymizes the features via: pixelating, blurring (Gaussian/motion) and masking techniques and sends the anonymized images to the adversary unit.

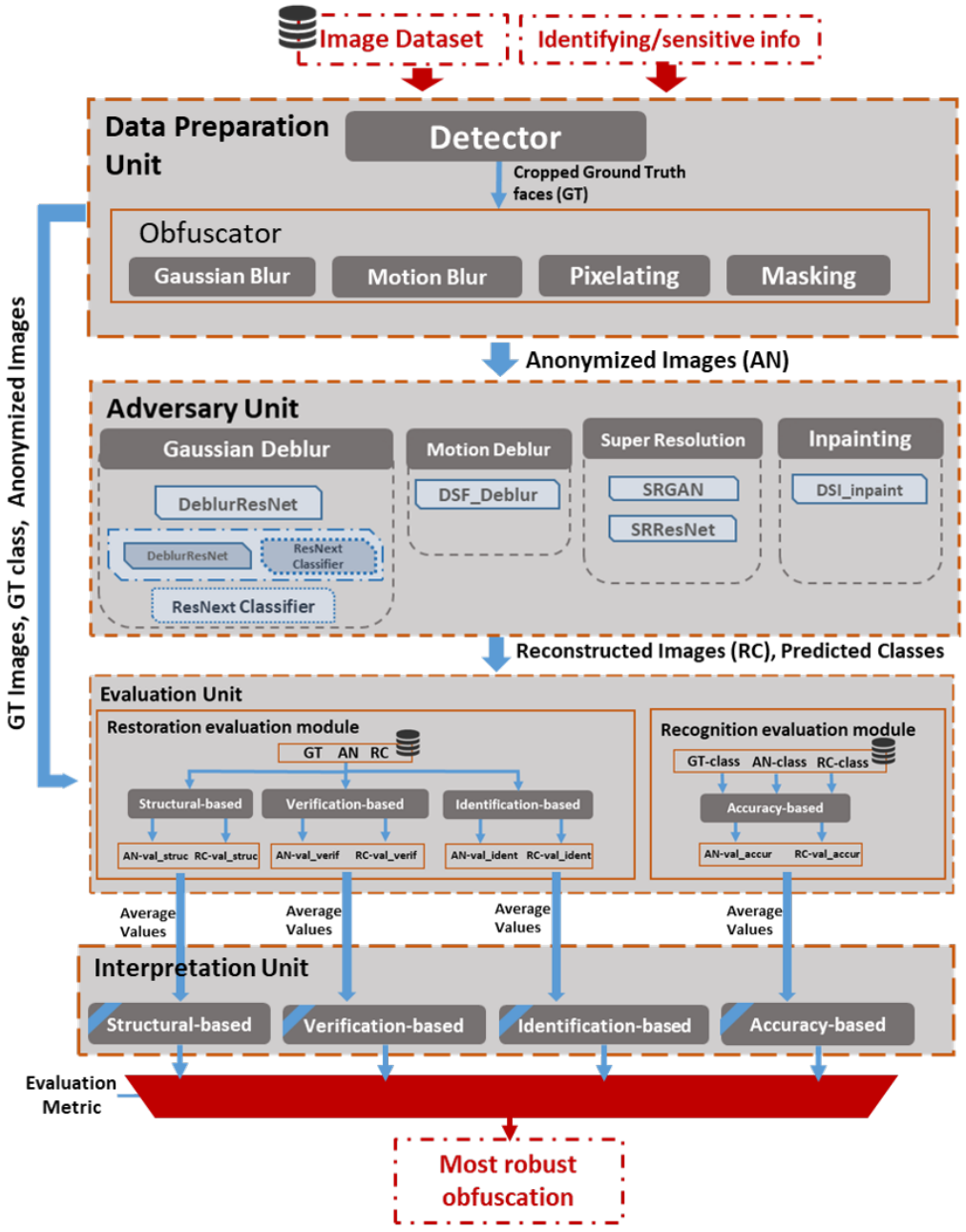


Figure 4.2: Adapting the proposed generic framework to images with faces as identifying features

4.2.2/ ADVERSARY UNIT

The adversary unit receives the obfuscated cropped face images. As shown in Figure 4.2, it is divided into four modules, one per obfuscation category: (a) the *super-resolution module* (for pixelating), (b) the *Gaussian deblur module* (for Gaussian blurring), (c) the *motion deblur module* (for motion blurring) and (d) the *inpainting module* (for masking). Each module contains one or more adversaries.

Adversary Model In our domain of application, an adversary undertakes an attack on obfuscated images in order to extract particular information from the hidden facial features. Inspired by the authors in [Do et al., 2018, Bellare et al., 1993a, Bellare et al., 1995a, Bellare et al., 2000], we define our adversary as a three-components model (c.f. Figure 4.3) :

- `Adversary's goal`: it refers to the adversary's intentions and to the particular information she/he attempts to obtain/extract from the anonymized dataset. In our work, the adversary's goal is to acquire the identity of the obfuscated faces.
- `Adversary's knowledge`: similar to [chapman & Hall Book, 2016], we differentiate between *external knowledge* and *background knowledge*. *External knowledge* is obtained from external sources (e.g. images gathered from public datasets, social network platforms...) whereas *background knowledge* is any sort of information regarding the anonymized dataset itself. In our work, (i) the obfuscation technique used to anonymize the target face images along with its hyperparameters and (ii) the identities $present$ in the target dataset constitutes the background knowledge.
- `Adversary's capabilities`: it represents to what extent can the adversary act in order to reach her/his goal, i.e. the *adversary's abilities*. It depends on the adversary's external and background knowledge. In our work, we consider that the adversary can perform a *restoration*-based, a *recognition*-based or a *R&R*-based attack.

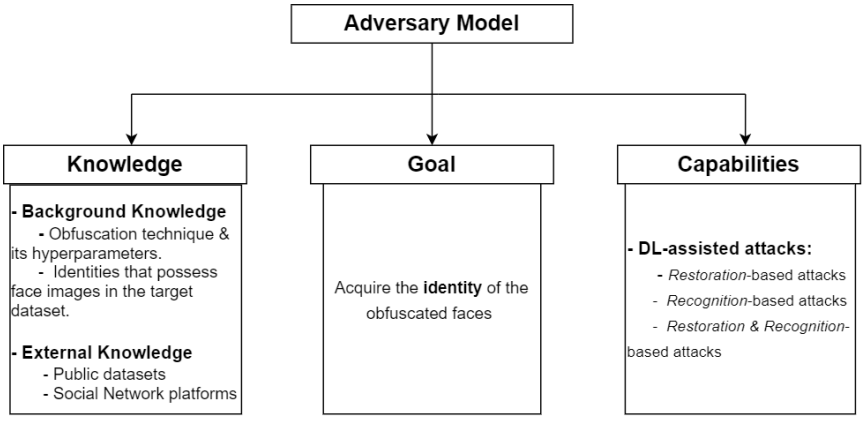


Figure 4.3: Adapting the 3-components adversary model to the face obfuscation scenario

Threat Levels Inspired by Shannon's Maxim[3], we defined three threat levels T_1, T_2 and T_3 with regard to the adversary's knowledge about the target obfuscated dataset (i.e., "*our system*") (c.f. Table 4.1):

[3] "*the enemy knows the system*", i.e, "*one ought to design systems under the assumption that the enemy will immediately gain full familiarity with them.*"

- `Threat level` T_1: assumes an adversary aware of the obfuscation technique used to protect the target dataset along with its hyperparameters.
- `Threat level` T_2: assumes an adversary with full/partial knowledge about the identities $present$ in the target dataset in addition to the obfuscation technique used and its hyperparameters.
- `Threat level` T_3: assumes an adversary with full/partial knowledge about the identities $present$ in the target dataset in addition to the obfuscation technique used without being aware of its hyperparameters.

Potential Attacks As we mentioned before, the adversary's capabilities scale with regard to her/his background knowledge. Hence, the attacks that the adversary can perform vary between T_1, T_2 and T_3. We consider three attacks: (i) *restoration*-based, (ii) *recognition*-based and (iii) *Restoration & Recognition*-based (*R&R*-based) attacks.

- `Restoration-based attack`: the adversary de-anonymizes obfuscated faces by trying to reconstruct the clear original features of the anonymized information. Training a Deep Neural Network to perform a *restoration*-based attack requires randomly gathering pairs of original/obfuscated face images. Hence, the adversary is capable of performing this sort of attack in T_1, T_2 and T_3[4].
- `Recognition-based attack`: the adversary breaches the images privacy and anonymity by training learning-based algorithms to perform recognition tasks on obfuscated faces. An identity *recognition*-based attack requires gathering obfuscated face images for specific identities. Hence, the adversary can perform this attack in T_2 and T_3.
- `Restoration & Recognition-based attack`: the adversary attempts to defeat the obfuscation technique via a two-steps attack: (1) reconstructing the hidden features of an obfuscated face and (2) trying to associate it with an identity by training an identity recognition model on clear face images. Therefore, only the adversaries in T_2 and T_3 can perform this two-steps process because it requires knowledge of the identities.

After each attack, the adversary outputs either a reconstructed face (in case she/he performed a *restoration*-based or a *R&R*-based attack) or a predicted class label/probability (in case she/he performed a *recognition*-based or a *R&R*-based attack). Both ways, each face image has three derivatives: the clear, obfuscated and reconstructed class/face as shown in Figure 4.4.

[4] In T_3 the *restoration*-based attack could be less dangerous compared to T_1 and T_2 because the adversary is not aware of the exact hyperparameters of the obfuscation technique.

Table 4.1: Comparing the adversary's capabilities and knowledge with regard to the two threat levels

Adversary's Components \ Threat Levels			T_1	T_2	T_3
Goal			**Identity/recover** the *identity* of the obfuscated faces		
Knowledge	**External Knowledge**	**Public Datasets**	✓	✓	✓
	Background Knowledge	**Obfuscation technique**	✓	✓	✓
		Obfuscation technique's hyperparameters	✓	✓	✗
		Identities *present* in the target dataset	✗	✓	✓
Capabilities	**DL-assisted attacks**	***Restoration*-based attack**	✓	✓	✓
		***Recognition*-based attack**	✗	✓	✓
		***Restoration* & *Recognition-based* attack**	✗	✓	✓

4.2.3/ EVALUATION UNIT

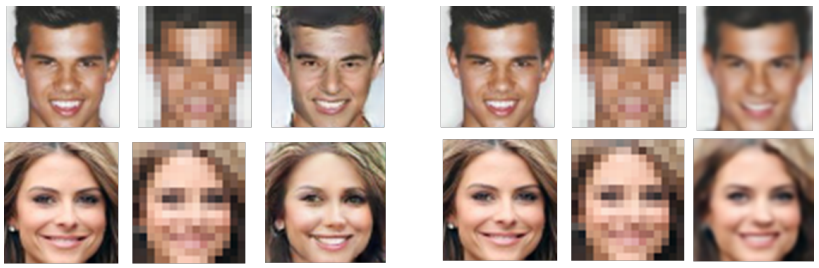
The evaluation unit is divided into two main modules: (1) the restoration and (2) the recognition evaluation modules (c.f. Figure 4.2). The former assesses the reconstruction ratio of the *restoration*-based attacks whereas the latter measures the accuracy of the *recognition*-based attacks. As for the *R&R*-based attacks, both the restoration and the recognition evaluation modules are employed.

Restoration evaluation module The restoration evaluation module assesses the face restoration with regard to *structural*, *verification* and *identification*-based metrics. Each metric-based sub-module receives as input three images: a clear face image (GT), an obfuscated face image (AN) and a reconstructed face image (RC).

– *The structural-based evaluation sub-module* quantifies the image enhancement/degradation quality after reconstruction attempts. In this study, we measure the holistic similarity between the clear image (GT) and the obfuscated image (AN) and between the clear image (GT) and the reconstructed image (RC) via SSIM[5] [Zhou Wang et al., 2004]. For normalization purposes, the *structural-based sub-module* computes the SSIM's complement, i.e. 1-SSIM. Hence, the output values are between 0 and 1 where 0 means the two images are identical:

$$AN\_value\_struc = 1 - SSIM(GT, AN) \tag{4.1}$$

[5]The Structural Similarity Index (SSIM) measures image quality modifications (enhancement/degradations)



(a) (b)

Figure 4.4: (a) Ground truth, Anonymized and Reconstructed images output via the SRGAN network. (b) Ground truth, Anonymized and Reconstructed image outputed via the SRResNet network.

$$RC\_value\_struc = 1 - SSIM(GT, RC) \tag{4.2}$$

– *The verification-based evaluation sub-module* validates the identity of a target face with a reference image. It mainly tries to conduct a 1-to-1 matching. In this study, we compute the identity distance via the OpenFace[6] toolbox [Amos et al., 2016] between the reference clear face images (GT) and both the obfuscated face image (AN) and the reconstructed face image (RC). OpenFace maps the two input faces to an identity distance between 0 and 4. The *verification-based evaluation module* normalizes the values to 0 and 1 where 0 value means that the two faces are identical, hence:

$$AN\_value\_verif = Normalized(OpenFace(GT, AN)) \tag{4.3}$$

$$RC\_value\_verif = Normalized(OpenFace(GT, RC)) \tag{4.4}$$

We employed in this study SSIM [Zhou Wang et al., 2004] and the OpenFace toolbox [Amos et al., 2016] because they are both publicly available and widely used in the literature to evaluate the reconstruction of degraded faces in the context of image transformation tasks [Shen et al., 2018a, Li et al., 2017a].

– *The identification-based evaluation sub-module* attempts to recognize an identifying feature of an individual based on a single face image. It mainly tries to compare the face in question with many others by conducting a 1-to-many matching. In our study, we employed a DL-based identity recognition model (c.f. Section 4.3.1). The *identification-based sub-module* uses the inferences over the three received images in order to compute two average relative error values[7]. The average relative error

[6] OpenFace is a Python and Torch implementation of face recognition with deep neural networks [Schroff et al., 2015]. OpenFace directly learns a mapping from face images to a compact euclidean space where distances directly correspond to a measure of face similarity.

[7] By definition, the average relative error is the absolute difference between the "exact theoretical" value

ranges between 0 and 1. We denote the class probability returned by the DL-based recognition model as *conf*.

$$AN\_value\_ident = \frac{|conf(GT) - conf(AN)|}{conf(GT)} \tag{4.5}$$

$$RC\_value\_ident = \frac{|conf(GT) - conf(RC)|}{conf(GT)} \tag{4.6}$$

In (5) and (6), both confidences in the numerator belong to the same *predicted class label*. In other words, in case the inferences of the recognition model over the obfuscated or reconstructed (*AN* or *RC*) image do not contain the GT class name, the *AN*_value_ident or the *RC*_value_ident would be 1.

AN-values and *RC*-values, outputted by the three restoration evaluation sub-modules, ranges between 0 and 1 where 0 indicates that the individual's privacy is completely breached whereas 1 means that it is intact. Each restoration evaluation sub-module computes the average *AN*-values and the *RC*-values over the entire obfuscated/restored dataset received from each DL-assisted attack and forwards them to the corresponding module in the interpretation unit.

Recognition evaluation module The recognition evaluation module assesses the face (obfuscated or restored) recognition ratio with regard to an *accuracy*-based metric. The *accuracy-based sub-module* receives as input the class names and probabilities predicted (by *recognition*-based or *R&R*-based attacks) over the obfuscated face image (AN-class) and the reconstructed face image (RC-class) along with the ground-truth label (GT-class).

– *Accuracy-based evaluation sub-module* measures the Top-n accuracy of the DL-based recognition models employed as *recognition*-based attacks by the adversaries. For each face image, the *accuracy-based sub-module* determines if the GT class label (GT-class) is equal to one of the top n predicted class labels[8] over the obfuscated (AN-class) and the reconstructed (RC-class) faces. After analyzing the entire obfuscated/restored dataset, the sub-module outputs the *AN-value_accur* and the *RC-value_accur*. For normalization purposes, we compute the fraction and the complement of the Top-n accuracy. Hence, the output values are between 0 and 1 where 0 means that the recognition model used by the adversary was highly accurate[9], i.e. the individual's anonymity is completely breached.

and its "measured" counterpart, divided by the "exact theoretical" value. We consider the inference over the clear face image (GT) as the "exact" value whereas the prediction over the anonymized (AN) and the reconstructed (RC) face images as the "measured" values.

[8] Class labels with the Top n highest probabilities.

[9] Top-n accuracy is 100%.

4.2.4/ INTERPRETATION UNIT

The interpretation unit selects the most robust obfuscation techniques per evaluation metric based on the results provided by the evaluation unit. As seen in Figure 4.2, the interpretation unit is divided into four *selection modules*, one per evaluation metric: (a) *structural-based*, (b) *identification-based*, (c) *verification-based* and (d) *accuracy-based selection module*. Each module performs a two-steps comparison in order to select the most resilient obfuscation technique: (1) *intra-attack* and (2) *inter-attack* comparisons (e.g., the *structural-based selection module* selects the most resilient obfuscation with regard to the SSIM metric whereas the *verification-based selection module* selects the most resilient obfuscation with regard to the Openface identity distance metric). As a first step, the *intra-attack* comparison allows us to identify the strongest DL-assisted attack against each obfuscation technique with regard to each evaluation metric. In other words, the attack that restored/recognized most of the obfuscated face images. As a second step, the *inter-attack* comparison chooses the most resilient obfuscation against the selected DL-assisted attacks. A detailed example is showcased in Section 4.3.1.

In this section, we described how our recommendation framework recommends the most resilient obfuscation technique via the 4-layered iterative workflow: (a) detecting/obfuscating the identifying/sensitive information, (b) restoring/recognizing via the DL-assisted attacks performed by the adversaries, (c) evaluating the reconstruction/recognition and (d) selecting the most robust obfuscation based on the inter/intra-attack comparisons.

4.3/ EXPERIMENTS

To validate and assess our approach, we set up our experiments to (i) evaluate the recommendation framework (c.f. Section 4.3.1) and study thoroughly the effect of the background knowledge on the adversary's capabilities with regard to (ii) the identities $present$ in the target dataset (c.f. Section 4.3.2) and (iii) the obfuscation technique (c.f. Section 4.3.3). Throughout the three experiments, we considered that the adversaries have the same goal: identify/recover the identities of the obfuscated faces. Furthermore, we conducted our experiments on the CelebA dataset [Liu et al., 2015] with the celebrity faces being the identifying features.

Table 4.2: The different adversaries considered for the first experimental setup

Adversary's Component \ Threat Levels			T_1			
			Adversary 1	**Adversary 2**	**Adversary 3**	**Adversary 4**
Goal			**Identity/recover** the *identity* of the obfuscated faces			
Knowledge	**External Knowledge**	**Public Datasets**	CelebA	CelebA	CelebA	CelebA
	Background Knowledge	**Obfuscation Technique**	Pixelation	Gaussian blurring	Motion blur	Masking
		Obfuscation Technique's hyperparameters	4x4	(31,31)		Random
		Identities known by the adversary	✗	✗	✗	✗
Capabilities	**DL-assisted attacks**	**Restoration-based attack**	SRGAN[36] SRResNet [25]	DeblurResNet[25]	DSF_Deblur[39]	DSI_inpaint[26]
		Recognition-based attack	✗	✗	✗	✗
		Restoration & Recognition-based attack	✗	✗	✗	✗

4.3.1/ EVALUATING THE RECOMMENDATION FRAMEWORK

In this experimental setup, we evaluate our recommendation framework by considering four obfuscation techniques: pixelation, Gaussian blur, motion blur and masking.

Input Dataset & Identifying Information In order to prepare our evaluation test dataset, we select[10] 370 face images from the official CelebA test set [Liu et al., 2015]. Our test set contains face images belonging to male and female celebrities of different races and different age (majority are above 18). To normalize our experimental setup, we use the same face images to evaluate the different DL-assisted attacks. The training sets vary between DL-assisted attacks, however, no face images from the test set were included throughout the training of any of the DL models.

Obfuscation techniques We employed in this setup four obfuscation techniques: (1) pixelation, (2) Gaussian blur, (3) motion blur and (4) masking. We specified for each obfuscation technique a fixed parameter as shown in Table 4.2. Regarding the pixelation, we simply downscaled the face images by a factor of 4. For the Gaussian blur, we applied a Gaussian filter with a kernel size (31,31) and standard deviation of 5. As for the motion blur, we synthesized a motion blur kernel from random 3D camera trajectories [Boracchi et al., 2012]. Regarding the masking technique, we replaced random

[10]We first selected 1307 images from the official CelebA test set, then we filtered out, via a pre-trained celebrity recognition model, the faces that were wrongly recognized or correctly recognized with a probability lower than 0.7. This recognition model was used by the *identification*-based evaluation sub module.

pixels all over the image by black pixels. As seen in Table 4.3, the different obfuscation techniques guarantee ”visually” the anonymity of the target identities.

Adversaries & DL-assisted attacks We simulated four adversaries in T_1 who perform 5 *restoration*-based attacks against the four obfuscation techniques (c.f. Table 4.2).

For the Super Resolution (SR) task, we considered that the adversary performed two *restoration*-based attacks against pixelation: SRResNet and SRGAN. On the one hand, the SRResNet is a ResNet-based architecture [He et al., 2016b] and is considered a benchmark when it comes to SR algorithms [Yang et al., 2019b, Ledig et al., 2017]. Moreover, SRResNet is a generic SR-network applicable to our faces dataset[11]. On the other hand, SRGAN is a GAN-based super resolution model implemented by [Garcia, 2016] similar to [Yu et al., 2016]. The model was developed specifically for faces. We generated the training pairs by downsampling the unobfuscated (GT) face images by a factor of four and trained both networks from scratch.

For the deblur task, we considered two distinct adversaries against two distinct blurring techniques. Regarding the Gaussian blur, we adapted the SRResNet architecture by modifying the input size of the network implemented in [Majumdar, 2016] (i.e., Deblur-ResNet). In addition, we generated the training pairs by applying Gaussian blur to the unobfuscated (GT) face images and trained the network from scratch. As for the motion blur, we used the implementation and the pre-trained model provided by the authors (i.e., DSF_Deblur) [Shen et al., 2018b].

Last but not least, we considered that the adversary applied the deep generative model DCGAN proposed in [Yeh et al., 2017] and the implementation in [Jin, 2018] to attack the masking technique (i.e., DSI_inpaint). We trained the DCGAN network on our face dataset from scratch. Table 4.3 summarizes the technical details regarding each DL-assisted attack.

Evaluation & Interpretation As stated in Section 4.2.3, the framework provides (i) structural, (ii) verification and (iii) identification-based evaluations to assess the reconstruction ration of the *restoration*-based attacks. Each evaluation sub-module in our framework computes two metric-based values for each clear image: (a) *AN*-value and (b) *RC*-value. These values range between 0 and 1 where 0 indicates that the individual's privacy is completely breached. In the following sections, we report the average values over the entire test set.

– *Structural-based evaluation:* As shown in Figure 4.5.(a), the average *RC_values_-*

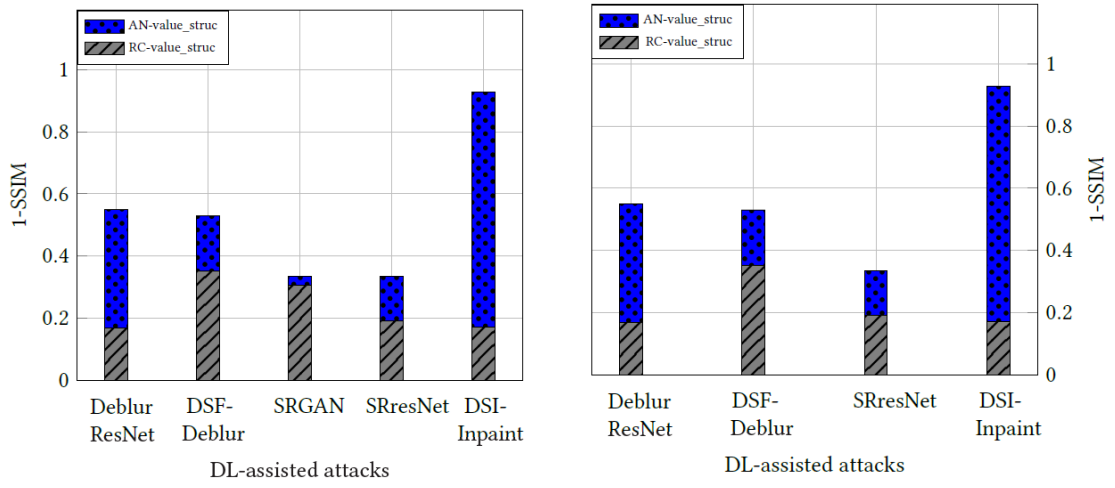
[11] The implementation [Majumdar, 2016] provided a network which upscales the input image by a factor of 2. Hence, we added an upscaling function and re-trained it from scratch for upscaling by a factor of 4.

Table 4.3: Technical details regarding the obfuscation techniques and the implementations of the DL-assisted attacks [Tekli et al., 2019]

Ground Truth face	Obfuscating technique	hyperparameters	Parameter values	Obfuscated faces	*Restoration*-based attacks	Implementation/ Framework	Results
	Pixelation	Pixel Box Size	4x4		**SRresNet** [25]	**TensorFlow** [38] Trained from scratch	
					SRGAN [36]	**TensorFlow** [36] Trained from scratch	
	Gaussian Blur	Kernel Size	(31,31)		**DeblurResNet** [25]	**Tensorflow** [38] Trained from scratch	
	Motion Blur	Length and angle of the motion			**DSF_deblur** [39]	**Matcaffe / Matlab** [39] Pre-trained model	
	Masking	Location of the black pixels	Random		**DSI_inpaint** [26]	**TensorFlow** [40] Trained from scratch	

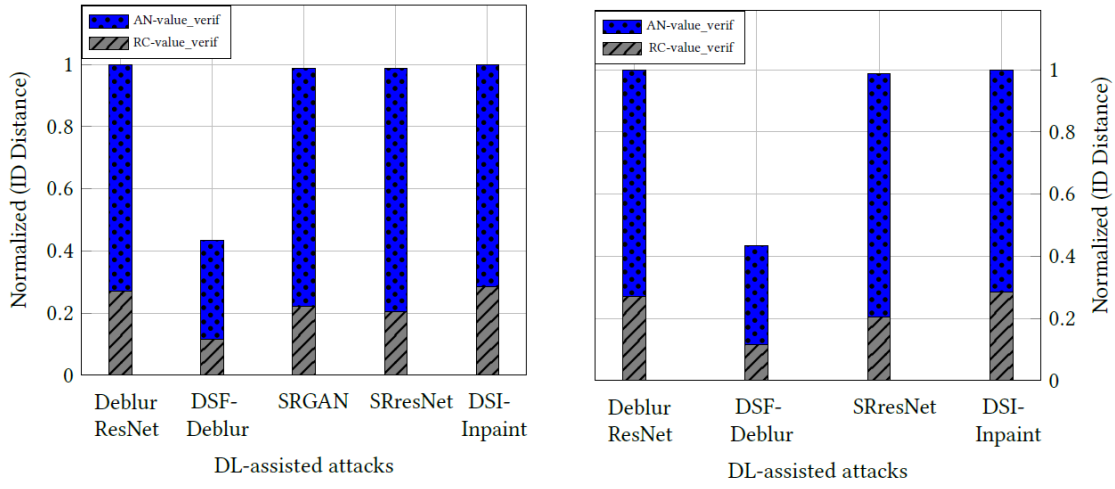
struct of all the DL-assisted attacks are lower than the average *AN_values_struct* since the reconstructed RC face images are overall more similar to the clear GT face images than the obfuscated AN face images in terms of SSIM. As mentioned in Section 4.2.4, the interpretation unit executes the intra/inter-attack comparisons in order to select the most resilient obfuscation. First, the intra-attack comparison selects the strongest DL-assisted attack against each obfuscation technique. For instance in our case, all adversaries performed a single DL-assisted attack against each obfuscation except "Adversary 1" (c.f. Table 4.2) which performed two DL-assisted attacks against the pixelation technique: (a) SRGAN and (b) SRResNet attacks. Therefore, the intra-attack comparison selects the attack that caused the highest privacy breach against pixelation, i.e. the SRResNet attack because it resulted in the lowest *RC_value_struct* as seen in Figure 4.5.(a) when compared to SRGAN. Furthermore, the inter-attack comparison selects the most resilient obfuscation technique, i.e. the obfuscation whose DL-assisted attack records the highest *RC_value_struct*. As seen in Figure 4.5.(b), the "DSF-Deblur" attack records the highest *RC_value_struc*, as such, "motion blur" is the most resilient obfuscation with regard to the SSIM metric.

– *Verification-based evaluation* In Figures 4.6.(a,b), we report the average *RC_val_verif* and *AN_val_verif* values. The intra/inter-attack comparisons select "masking" as the most resilient obfuscation technique with regard to the identity distance metric because the corresponding "DSI_Inpaint" attack recorded the highest *RC-value_verif* in Figure 4.6.(b).



(a) Before *Intra*-attack comparisons (b) After *Intra*-attack comparisons

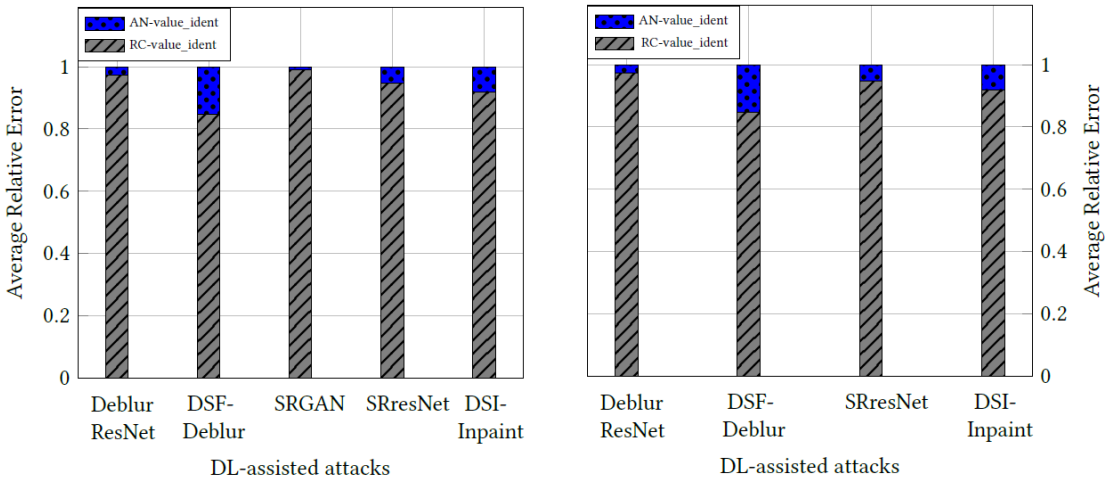
Figure 4.5: The *Structural-based evaluation sub-module* output before and after the intra-attack comparisons [Tekli et al., 2019]



(a) Before *Intra*-attack comparisons (b) After *Intra*-attack comparisons

Figure 4.6: The *Verification-based evaluation sub-module* output before and after the intra-attack comparisons [Tekli et al., 2019]

– *Identification-based evaluation* In Figures 4.7.(a,b), we report the average *AN_values_ident* and *RC_values_ident*. The intra/inter-attack comparisons select "Gaussian blur" as the most resilient obfuscation technique with regard to the identification-based metric because the "DeblurResNet" attack recorded the highest *RC_values_ident* in Figure 4.7.(b).



(a) Before *Intra*-attack comparisons (b) After *Intra*-attack comparisons

Figure 4.7: The *Identification-based evaluation sub-module* output before and after the intra-attack comparisons [Tekli et al., 2019]

The ”masking” technique would be the most robust obfuscation with regard to the different metrics, even after reconstruction attempts, if we were to block the entire face image with black pixels. In this study, we are masking the face image by randomly placing black pixels and leaving some original pixels intact (c.f. Figure 2.8.(e)), hence our results.

Comparison with other evaluation frameworks On a different note, the authors in [Li et al., 2017b] considered a human-based evaluation where they showed each participant an obfuscated face image and a couple of clear face images in order for her/him to match them up and guess the obfuscated identity. If we were to apply the same scenario to our four obfuscations, it would be more difficult for a human to re-identify an identity masked with random black pixels in comparison to the pixelated or blurred identities (as seen via the obfuscated face images in Figure 4.8). Nevertheless, after reconstructing the obfuscated faces via *restoration*-based attacks, the ”masking” technique becomes also vulnerable to the human visual system (as seen via the reconstructed face images in Figure 4.8). Similarly, if we were only to evaluate the obfuscation techniques like the authors did in [Nawaz et al., 2017, Dufaux et al., 2010, Korshunov et al., 2013] where they compared the obfuscated image to the original image via quantitative metrics, then ”masking” would be the most resilient obfuscation with regard to the different evaluation metrics as we notice when observing the AN-values in Figures 4.5, 4.6 and 4.7 (e.g. the AN-values of the ”masking” technique are always 1). However, we notice that it is not always the case after performing the *restoration*-based attacks: for instance when observing Figure 4.5.(b), we notice that the *RC_value_struct* of ”masking” is lower than ”pixelation” and ”motion blur”, i.e. the reconstruction of the masked face images was better with regard to the SSIM metric compared to the pixelated and motion blurred face images therefore

making it more vulnerable. These two observations stress the importance of employing *restoration*-based attacks when evaluating the robustness of an obfuscation technique.

As a future step, we would like to add a *human-based evaluation sub-module* similar to what the authors did in [Li et al., 2017b]. Therefore, we would have the possibility to select the most resilient obfuscation technique after reconstruction attempts with regard to the human visual system as well.

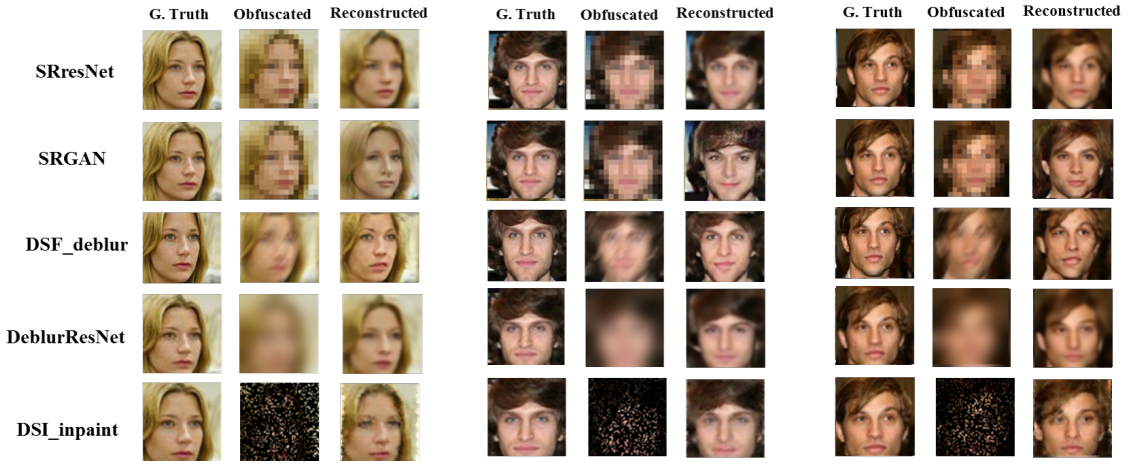


Figure 4.8: Comparison of the different reconstructions. Columns from left to right include Ground truth, Obfuscated and Reconstructed faces. Rows from top to bottom include the DL-assisted attacks [Tekli et al., 2019]

4.3.2/ STUDYING THE EFFECT OF THE BACKGROUND KNOWLEDGE REGARDING THE IDENTITIES *present* IN THE TARGET DATASET

In this experimental setup, we show how the adversary's knowledge regarding the identities *present* in the target dataset affects her/his capabilities. In the earlier section, we identified "masking", "motion blur" and "Gaussian blur" as the most robust obfuscations with regard to the different evaluation metrics in our framework. In this section, we focus on the "Gaussian blur" obfuscation technique as it is still the most widely used technique for privacy preservation purposes [Frome et al., 2009, Caesar et al., 2020, Yang et al., 2021].

Data Preperation In this setup, we needed to train identity recognition models in order to perform *recognition*-based and *R&R*-based attacks. Hence, we had to gather face images for each identity. Although the CelebA dataset [Liu et al., 2015] is not designated for identity recognition tasks, we used it for training and evaluating the DL-assisted attacks[12].

[12]For instance, we did not employ the FaceScrub dataset [Ng et al., 2014], which is designated for identity recognition tasks, because the number of identities is limited to 530 whereas it is 10,177 in the CelebA dataset.

We selected[13] 854 identities from the CelebA dataset and we gathered 60 face images for each celebrity[14]. Out of these 60 images, 5 were left for testing and the remaining 55 were used for training purposes. Therefore, our test set contained 4270 face images (854 selected individuals x 5 test images) which are not part of the official CelebA test set (58% are female and 42% are male). We resized all the images to 64x64 and then applied the blurring function with a kernel size (31x31) and standard deviation of 5 (c.f. Figure 2.8.(c)).

Incremental Background Knowledge In order to simulate the adversary in T_2, we designed an adversary with an incremental background knowledge regarding the number of identities $present$ in the target dataset. We denoted as N the set of identities known by the adversary. We varied $|N|$ between 0 (no knowledge about the identities $present$ in the target dataset, i.e. T_1) and 854 (Full knowledge, i.e. T_2)[15]. In total, we considered 10 distinct values for $|N| = \{0, 100, 200, ..., 800, 854\}$.

Adversary & DL-assisted attacks As mentioned in Section 4.2.2, the adversary in T_2, is capable of executing either a (i) *R&R*-based or a (ii) *recognition*-based attack.

On the one hand, the *R&R*-based attack is a combination of a DL restoration model followed by a DL identity recognition model. Regarding the restoration model, we trained the same DeblurResNet network [Ledig et al., 2017, Majumdar, 2016] as in section 4.3.1. The only difference is that we included in the training set 10 pairs of clear/obfuscated face images for each identity in N. As for the recognition model, the adversary uses the remaining 45 clear face images of each identity in N to train a SEResNext101[16] classifier with $|N| + 1$ classes[17] and attempt to recognize reconstructed faces [Xie et al., 2017, Hu et al., 2019].

On the other hand, the *recognition*-based attack tries to associate each anonymized face image with an identity (bypassing the reconstruction process). Therefore, the adversary obfuscates the 55 face images of each identity in N and train a SEResNext101-based classifier with $|N| + 1$ classes in order to recognize obfuscated face images. We applied transfer learning to our classifier network by employing ImageNet pre-trained weights [Russakovsky et al., 2015b] and also augmented our training datasets by randomly flipping, resizing and adding noise (e.g. color variations and saturation) to the face images.

We simulated for each value of $|N|$ the corresponding attacking capabilities hence we

[13] for additional details regarding the data preparation process, please contact jimmytekli@hotmail.com
[14] we mined images from google via *google-images-download* as well.
[15] 854 being the maximum number of individuals in our test set
[16] https://github.com/BMW-InnovationLab/BMW-Classification-Training-GUI
[17] In addition to the classes regarding the individuals in N, we also added an additional class to our classifier entitled “others” which grouped 800 images that belong to other individuals

trained 1 *restoration*-based, 9 *R&R*-based and 9 *recognition*-based attacks as seen in Table 4.4.

Table 4.4: Technical details regarding the DL-based models employed as *restoration*, *recognition* and *R&R*-based attacks

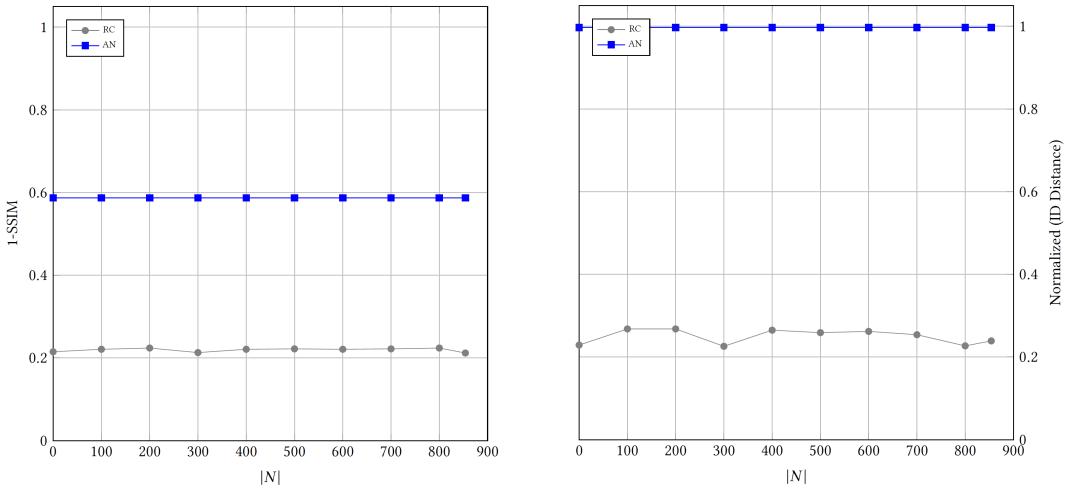
Adversary's Component \ Threat Levels			T_1	T_2			
Goal			**Identity/recover** the *identity* of the obfuscated faces				
Knowledge	**External Knowledge**	**Public Datasets**	**CelebA & Google Images** which resulted to **55** face images for each identity in N				
	Background Knowledge	**Obfuscation Technique & hyper-parameters**	Gaussian Blurring (31,31)				
		Set of identities N known by the adversary	0	100	200	…	854
Capabilities	**DL-assisted attacks**	***Restoration*-based attack**	DeblurResNet [18] trained on randomly chosen face images from CelebA dataset.	×	×	…	×
		***R&R*-based attack**	**Restoration task** ×	**Restoration task** DeblurResNet trained with a dataset that includes **10** face images of each of the **100** identities in **N**	**Restoration task** DeblurResNet trained with a dataset that includes **10** face images of each of the **200** identities in **N**	…	**Restoration task** DeblurResNet trained with a dataset that includes **10** face images of each of the **854** identities in **N**
			Recognition task ×	**Recognition task** SEResNext101 Classifier with **101** classes where the first **100** classes contain **45** clear face images of each identities in N and the additional class contains 800 face images of other identities	**Recognition task** SEResNext101 Classifier with **201** classes where the first **200** classes contain **45** clear face images of each identity in N and the additional class contains 800 face images of other identities	…	**Recognition task** SEResNext101 Classifier with **855** classes where the first **854** classes contain **45** clear face images of each identity in N and the additional class contains 800 face images of other identities
		***Recognition*-based attack**	×	SEResNext101 Classifier with **101** classes where the first **100** classes contain **55** obfuscated face images of each identity in N and the additional class contains 800 face images of other identities	SEResNext101 Classifier with **201** classes where the first **200** classes contain **55** obfuscated face images of each identity in N and the additional class contains 800 images of other identities	…	SEResNext101 Classifier with **855** classes where the first **854** classes contain **55** obfuscated face images of each identity in N and the additional class contains 800 face images of other identities

Results and Interpretations We show how the incremental background knowledge with regard to the identities $present$ in the target dataset affect the adversary's capabilities. Our results show that:

- The incremental background knowledge does not affect the reconstruction accuracy of the restoration models in the *R&R*-based attacks.
- The adversary breaches the privacy of the obfuscated face images via both the *R&R*-based and the *recognition*-based attacks.
- The incremental background knowledge increases the accuracy of the recognition models in both the *R&R*-based and the *recognition*-based attacks, i.e. increases the privacy breaches.
- The adversary is more dangerous when performing *recognition*-based attacks compared to *R&R*-based attacks.

As stated in Section 4.2, our framework provides structural and verification-based evaluations regarding the *restoration*-based attacks. In the following part, we measure the AN-values and the RC-values of the restoration models in the *restoration*-based and *R&R*-based attacks for each value of $|N|$.

- **The incremental background knowledge does not affect the reconstruction accuracy of the restoration models in the *R&R*-based attacks**: We notice in Figure 4.9 for the different values of $|N|$ that (a) the *RC_values_struct* and (b) *RC_values_verif* values are stable with minor fluctuations. This demonstrate that increasing $|N|$ does not increase nor affect the reconstruction accuracy of the restoration models with regard to the SSIM [Zhou Wang et al., 2004] and the Openface [Amos et al., 2016] evaluation metrics. In other words, even if the adversary knows the identity of a particular individual in the target dataset, adding face images of this particular individual to the training set of the restoration model (DeblurResNet [Majumdar, 2016] in our case) does not affect its reconstruction accuracy.



(a) *Structural*-based metric (b) *Verification*-based metric

Figure 4.9: Effect of the background knowledge on the reconstruction quality with regard to the structural and verification-based evaluation sub-modules

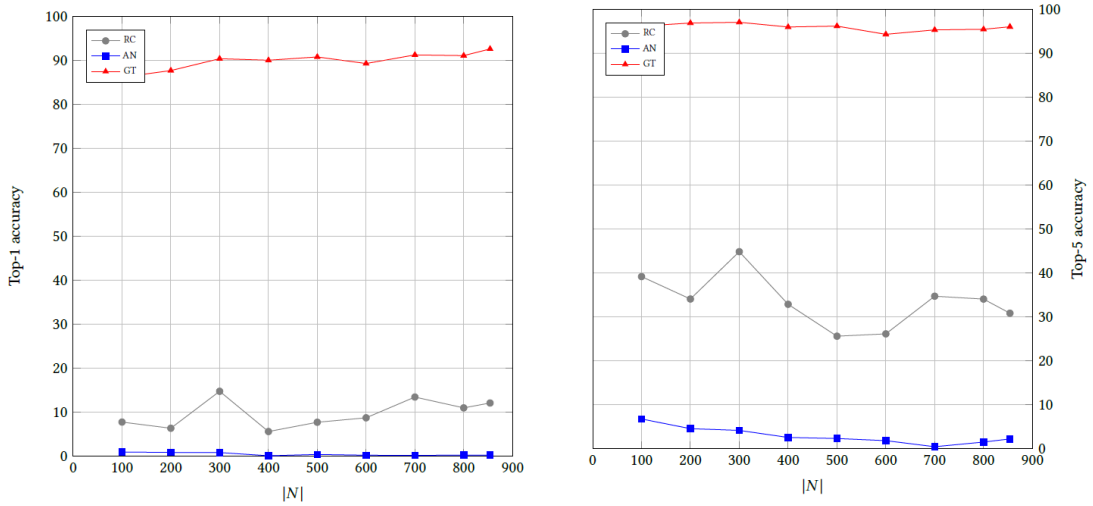
In the following part, we measure the Top-1 (Top-5) accuracy of the identity recognition models in the *R&R*-based attacks (SEResNext101 classifiers trained on clear faces to recognize reconstructed faces) and in the *recognition*-based attacks (SEResNext101 classifiers trained on obfuscated faces to recognize obfuscated faces).

- **The adversary breaches the privacy of the obfuscated face images via both the *R&R*-based and the *recognition*-based attacks.**

  In Figures 4.10.(a,b), we show the Top-1(Top-5) accuracy of the identity recognition models employed in the *R&R*-based attacks for each value of $|N|$. Therefore,

we report in both figures three distinct values: the GT values[18] which serve as a reference, the RC[19] and the AN[20] values in order to highlight the effect of the face reconstruction process on the classifier's accuracy compared to the anonymized face images. We notice that Top-1(Top-5) accuracies report higher values for the RC curve in comparison with the AN curve by almost 10% (30%). In other words, the adversary breached and recovered, via *R&R*-based attacks, the identity of the obfuscated faces 10(30) times more after reconstruction. The slight variations observed in the RC curves are due to the margin of error resulted from the *restoration*-based attack.

Furthermore, we report in Figures 4.11.(a,b) the Top-1(Top-5) accuracy of the identity recognition-models employed as *recognition*-based attacks by the adversary. In this case, we only report the GT and the AN values without the RC values because the reconstruction process is not part of the attack. The Top-1(Top-5) accuracy reports higher values for the AN curves when compared to the GT curves because the identity recognition models are trained on obfuscated face images in this case. Most importantly, we notice that the adversary breached and recovered, via the *recognition*-based attacks, the identity of the obfuscated faces nearly 55%(72%) of the time.

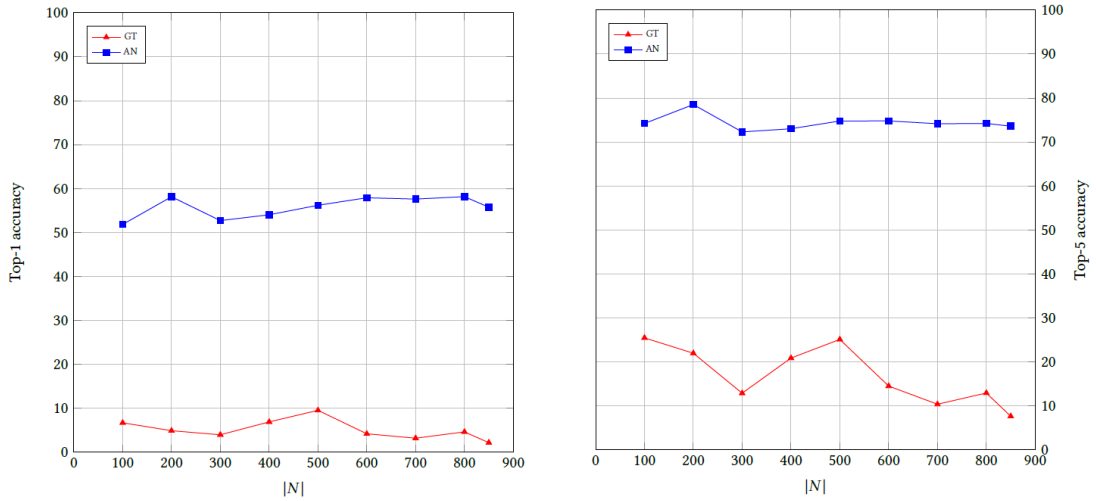


(a) Top-1 accuracy

(b) Top-5 accuracy

Figure 4.10: Measuring the Top-1/Top-5 accuracy of the different *R&R*-based attacks for each value of $|N|$

[18]The accuracy of the classifiers when inferring over GT face images
[19]The accuracy of the classifiers when inferring over RC face images
[20]The accuracy of the classifiers when inferring over AN face images



(a) Top-1 accuracy (b) Top-5 accuracy

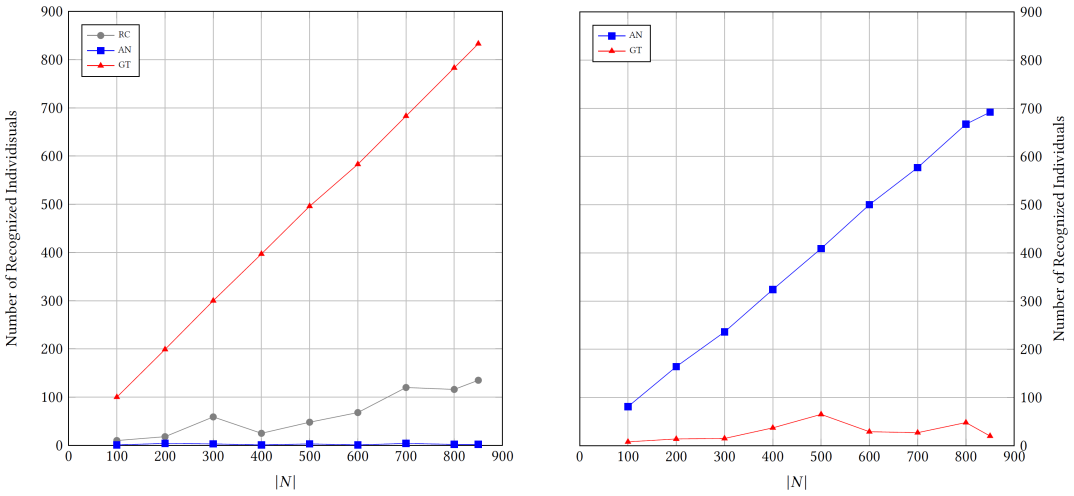
Figure 4.11: Measuring the Top-1/Top-5 accuracy of the different *recognition*-based attacks for each value of $|N|$

In the following part, we count the number of individuals who were re-identified and whose anonymity was breached. As we mentioned before, our test set contains 5 anonymized face images per identity. Hence, we consider that an individual is re-identified if L face images out of 5 are correctly recognized (Top-1 recognition) where $0 < L <= 5$. In the following, we report the values for $L = 2$.

- **The incremental background knowledge increases the accuracy of the recognition models in both the *R&R*-based and the *recognition*-based attacks, i.e., increases the privacy breaches:** In Figure 4.12.(a), we count the number of individuals re-identified by the *R&R*-based attacks. Because the recognition model is trained on clear face images, the GT curve serves as a reference. We report that the number of re-identified individuals with regard to the RC images increased along with the background knowledge of the adversary. For $|N|$=100, the adversary re-identified, via the *R&R*-based attack, 10 out of 854 (1.2%) individuals after reconstruction whereas at $|N|$=854 she/he recognized 135 individuals (15.8%). In addition, in Figure 4.12.(b) we report the number of individuals re-identified by the *recognition*-based attacks. We notice a steady increase in the number of re-identified individuals with regard to the AN face images along with the background knowledge. At $|N|$=854, the adversary re-identified 692 individuals out of 854, i.e., almost 81% of the anonymized individuals. The *recognition*-based attacks demonstrate poor results when inferring over clear (GT) face images in Figure 4.12.(b) because the identity recognition models are trained via obfuscated face images.

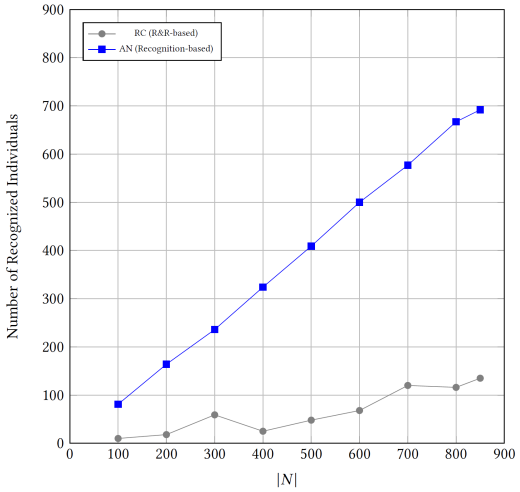
- **The adversary is more dangerous when performing *recognition*-based attacks compared to the *R&R*-based attacks:** When comparing the adversary's capa-

bilities in Figure 4.12.(c), we notice that when equipped with $|N|$=100 as background knowledge, the adversary re-identified 10 individuals when performing a *R&R*-based attack whereas she/he re-identified 81 when performing a *recognition*-based attack. The same behavior persists throughout the incremental process of the background knowledge. When equipped with $|N|$=854 as background knowledge, the adversary re-identified 135 out of 854 (15.8%) when performing an *R&R*-based attack whereas she/he re-identified 692 (79.8%) when performing a *recognition*-based attack.



(a) *R&R*-based attacks

(b) *Recognition*-based attacks



(c) *Recognition*-based vs *R&R*-based attacks

Figure 4.12: Counting and comparing the number of recognized individuals in the test set when performing *R&R*-based and *recognition*-based attacks

4.3.3/ STUDYING THE EFFECT OF THE BACKGROUND KNOWLEDGE REGARDING THE OBFUSCATION TECHNIQUE

In this experimental setup, we show how the background knowledge with regard to the obfuscation technique and its hyper-parameters affects the adversary's capabilities. Similar to the previous section, we consider "Gaussian blur" as the obfuscation technique.

Data Preparation We selected randomly 100 identities from the dataset prepared in Section 4.3.2. For each identity, 55 face images were left for training purposes and 5 images were left for testing (i.e. 500 images in the test set). We prepared 7 different versions of the target dataset, each blurred with a kernel from k_{test}= {19, 25, 31, 37, 43, 49, 55} (c.f. Table 4.5).

Table 4.5: The seven target datasets blurred with distinct k_{test} values

Original	(19,19)	(25,25)	(31,31)	(37,37)	(43,43)	(49,49)	(55,55)
							

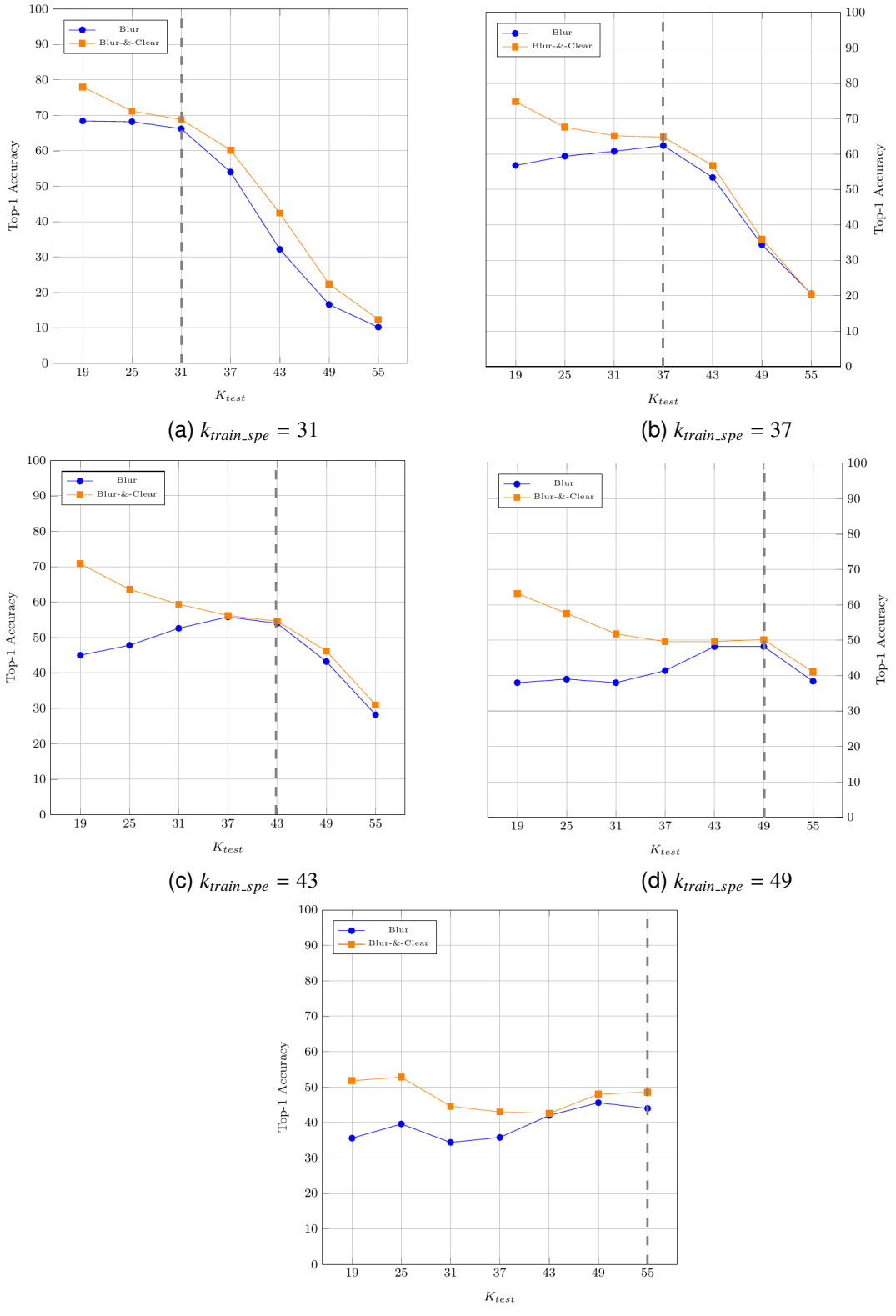
Background knowledge We consider threat level T_3, i.e., the adversary is aware of the obfuscation technique employed in the test/target dataset (e.g., Gaussian blur) however not of its hyper-parameters (e.g., the blurring kernel's size). In addition, we consider the adversary is aware of the identities in the target dataset (i.e., $|N| = 100$).

Adversary & DL-assisted attacks We perform *recognition*-based attacks as it is more dangerous compared to the *R&R*-based attacks as demonstrated in the earlier section (c.f. Section 4.3.2). We employ the same SEResNext101 classifier and training parameters used in section 4.3.2. In T_3, the adversary can choose any blurring kernel and prepare the training dataset accordingly because she/he is not aware of the blurring kernel used to obfuscate the target dataset. Hence, we trained 5 *recognition*-based attacks, each with a distinct kernel from k_{train}={31,37,43,49,55}. We report the privacy breaches of each attack against the 7 target datasets blurred via k_{test}. Last but not least, we considered two training modes for each *recognition*-based attack: (i) "**blur-&-clear mode**" where the training set contains clear and blurred version of each face image and (ii) "**blur mode**" where the training set contains blurred face images only.

Results and Interpretations In the following section, we demonstrate that:

- The adversary must not know the exact blurring kernel of a target dataset in order to breach its anonymity.
- The privacy breaches decrease steadily in a linear fashion when attacking face images blurred with kernels greater than the kernel chosen by the adversary while preparing her/his training dataset.
- Including both, clear and blurred images in the training datasets increases the recognition accuracy of the *recognition*-based attacks, specifically when the target dataset's blurring kernel is smaller than the training dataset's.
- Preparing the training dataset with blurring kernel (37,37) provides the widest attack range against the 7 target datasets.

Each subfigure in Figure 4.13 reports the Top-1 accuracy of the *recognition*-based attacks (**blur-&-clear** and **blur** modes) trained with a specific kernel k_{train_spe} and attacking the 7 target datasets. For instance, Figure 4.13.(a) corresponds to the *recognition*-based attacks (**blur-&-clear** and **blur** modes) trained on face images blurred via k_{train_spe}=(31,31). We report the following observations:



(a) $k_{train\_spe} = 31$

(b) $k_{train\_spe} = 37$

(c) $k_{train\_spe} = 43$

(d) $k_{train\_spe} = 49$

(e) $k_{train\_spe} = 54$

Figure 4.13: Top-1 accuracy of the *recognition*-based attacks (**blur-&-clear** and **blur** modes) trained with a specific kernel k_{train_spe} and attacking the 7 target datasets.

- **The adversary must not know the exact blurring kernel of the target dataset in order to breach its anonymity.** For instance in Figure 4.13.(e), we notice that the adversary breached the faces' anonymity 46% (**blur-&-clear** mode) and 36% (**blur** mode) of the time on average against the 7 target datasets despite training the recognition models with kernel size k_{train_spe}=(55,55). Similar behavior is observed, although with different magnitudes, for the other k_{train_spe} values as well.

- **The privacy breaches decrease steadily in a linear fashion when attacking face images blurred with kernels greater than k_{train_spe}.** As we notice in Figure 4.13.(a), the Top-1 accuracies (for both **blur-&-clear** and **blur** modes) decrease steadily when the adversary attacks target datasets blurred with kernels greater than k_{train_spe}=(31,31). The same behavior clearly persists in Figures 4.13.(b) and 4.13.(c) for k_{train_spe}=(37,37) and k_{train_spe}=(43,43) respectively.

- **Including both, clear and blurred images in the training datasets increases the recognition accuracy of the *recognition*-based attacks, specifically when attacking a target dataset obfuscated with a kernel smaller than k_{train_spe}.** When trained via the **blur** mode, the highest privacy breach occurs when the training and the target datasets are blurred with the same kernel, i.e., k_{train_spe}= k_{test}. Whereas, when trained via the **blur-&-clear** mode, the highest privacy breach occurs against the target dataset blurred with the smallest kernel, i.e., k_{test}=(19,19). Last but not least, we notice that both, **blur** and **blur-&-clear** modes report almost the same Top-1 accuracies for k_{test}=k_{train_spe} when observing subfigures 4.13.(b)-(c) and (d).

- **Using blurring kernel (37,37) provides the widest attack range and highest privacy breaches against the 7 target datasets.** To estimate the range of each attack against the 7 target datasets, we report in Table 4.6 the Area Under the Curve (AUC) of the Top-1 accuracy curves for each kernel in k_{train} (i.e., for each sub-figure in Figure 4.13). We notice that kernel size (37,37) reports the highest AUC values for both training modes, i.e. the highest privacy breaches and attack range against the 7 target datasets. The adversary does not need to blur her/his training datasets with the highest blurring kernel (k_{train_spe}=55 in our case) in order to cause the highest privacy breaches over the different target datasets. In other words, considering an adversary unaware of the target dataset's blurring kernel, the most dangerous attack she/he could perform is a *recognition*-based attack trained via the **blur-&-clear** mode using kernel k_{train_spe}=(37,37).

Table 4.6: AUC table

Mode \ k_{train_spe}	(31,31)	(37,37)	(43,43)	(49,49)	(55,55)
Blur	276.5	**309.1**	290	253	237.2
Blur-&-Clear	310.2	**338**	330.9	310.9	281.2

4.4/ FRAMEWORK DISCUSSION

In this section, we describe briefly how our recommendation framework is generic in terms of identifying/sensitive features and scalable with regard to the obfuscation techniques, the adversaries and the evaluation metrics.

First, our recommendation framework is generic because it can be adapted to other identifying/sensitive information. Let us consider the worker's badge name as the identifying feature instead of the worker's face. In other words, our goal is to recommend the most robust obfuscation techniques for the workers' badge names. In short, we need to:

- Change/train a detector in the data preparation unit to localize and detect the text/Badge Names in an image, e.g. OpenCV's scene text detector[21].
- Train DL-assisted attacks to restore/recognize obfuscated characters in image by adding pairs of clear/obfuscated text images to their training sets.
- Change/adapt the evaluation metrics in the evaluation unit: for instance, employing the Tesseract OCR library[22] in the verification-based sub-module in order to extract the text from the clear, obfuscated and restored images and compare the extracted results.

Second, our recommendation framework is scalable with regard to the:

- *Obfuscation techniques*: we can evaluate the robustness of the GAN-based inpainting method in the context of face images [Hao et al., 2019] by implementing it in the obfuscater (data preparation unit) and train a DL-assisted attack accordingly.
- *Adversaries*: we can also consider adversaries with different threat levels, capable of performing more dangerous DL-assisted attacks either by considering additional knowledge, different neural network architectures [Zhang et al., 2020], other training hyper-parameters or larger training datasets...
- *Evaluation metrics*: including additional metrics provides 'redundancy' and 'diversity' for the evaluation process. For instance in the context of face images, we can consider human evaluators as in [Li et al., 2017b] or other identity-based metric to measure the identity distance between two faces alongside the OpenFace tool.

[21] https://docs.opencv.org/master/da/d56/group__text__detect.html
[22] https://github.com/tesseract-ocr/tesseract

4.5/ RELATED WORKS

In this section, we investigate works related to (i) adversaries attacking obfuscated images via *recognition*/*restoration*-based attacks and (ii) to evaluation frameworks.

4.5.1/ RECOGNITION-BASED ATTACKS

In [Newton et al., 2005b], Newton et al. designed an algorithmic attack to identify people from pixelated and blurred face images. The recognition rates increased after applying the same obfuscation to the probe and gallery set of the face recognition approach [Turk et al., 1991]. They showed that small pixel box (e.g., 2x-4x) and simple blurring cannot prevent identification attacks. In another study [Gopalan et al., 2012], Gopalan et al. presented a method to recognize faces obfuscated with non-uniform blurring by examining the blurred images. As a follow-up study [Punnappurath et al., 2015], Punnappurath et al. applied blurring effects to images in the target gallery and measured the minimal distance between the gallery images and the blurred probe image. On another note, the authors in [McPherson et al., 2016] demonstrated that modern image recognition approaches, based on artificial neural networks, can be employed as attacks to recover hidden information from obfuscated images. They focused on three forms of obfuscation: pixelating, blurring and P3 (an encryption-based method [Ra et al., 2013]). The adversary successfully identifies obfuscated faces and objects by training DL networks with obfuscated images (faces [Ng et al., 2014, Cambridge, 1994], digits [LeCun, 1998] and objects [Krizhevsky et al., 2009]). Also in a recent medical study [Packhäuser et al., 2021], the authors performed DL-assisted attacks against a publicly available anonymized medical dataset [Wang et al., 2017] containing x-rays of patients with sensitive meta-data such as treatment history, clinical institution, diagnosis... They considered an adversary aware of the identities *present* in the target dataset. Therefore, she/he can perform *recognition*-based attacks and link the known identity to the anonymized x-rays in the target anonymized dataset in order to gain more sensitive data about the identity.

Similar to [Packhäuser et al., 2021] and unlike the other studies, we assume a more realistic scenario where an adversary can perform a *recognition*-based attack only when equipped with the proper background knowledge. Additionally in our case, we study thoroughly how the background knowledge affects the *recognition*-based attacks. For instance, in T_2 we show how the incremental background knowledge regarding the identities *present* in the target dataset intensifies the privacy breaches and increases the number of re-identified individuals. Whereas in T_3, we show how an adversary can perform a *recognition*-based attack and breach the face's anonymity despite lacking knowledge regarding the hyper-parameters of the obfuscation technique used.

4.5.2/ RESTORATION-BASED ATTACKS

The authors in [Ruchaud et al., 2016] tackled the privacy-preservation question in the context of obfuscated faces by restoring obfuscated features and evaluating the reconstruction with regard to face recognition. They considered three obfuscations: pixelating, blurring and masking. They used traditional image reconstruction techniques (i.e., reconstruction [Dong et al., 2011] and interpolation-based [Keys, 1981] techniques for super resolution). In addition, they evaluated the identity restoration using the same traditional face recognition techniques as in [Korshunov et al., 2013]. In our framework, we adopted DL-based techniques for both, face reconstruction and recognition because as stated in [Ledig et al., 2017, Amos et al., 2016], DL-based techniques demonstrate great superiority over traditional methods. Alternatively, the authors in [Abramian et al., 2019] investigated the amount of obfuscation needed to guarantee patients anonymity. They applied CycleGAN [Zhu et al., 2017] in order to reconstruct features from anonymized medical imaging. They considered two anonymization techniques: (a) blurring and (b) masking. They also compared the results qualitatively and quantitatively by computing correlation coefficients and SSIM between the original and reconstructed images as well as between the original and anonymized images. In our approach, we add a level of abstraction to the restoration and evaluation process, i.e. the intra/inter attack comparisons in the interpretation unit, in order to not only evaluate the reconstruction process but recommend the most robust obfuscation technique.

4.5.3/ BACKGROUND KNOWLEDGE EFFECT

In a similar study to ours, the authors in [Hao et al., 2019] evaluated the effectiveness of 8 obfuscation techniques by considering three threat levels based on the knowledge of the adversary with regard to the obfuscation technique employed along with its hyper-parameters. They considered that the weakest adversary has no knowledge about the obfuscation used whereas the strongest knows the exact one employed. In addition, they performed three types of attacks: an *recognition*-based, a *verification*-based and a *restoration*-based attack and they showed that the privacy breaches increase along with the background knowledge. In our work, we first defined the background knowledge of the adversary with regard to the identities $present$ in the target dataset, not only the obfuscation technique employed. Second, we designed the adversary with an incremental background knowledge with regard to the number of identities known by the adversary. Whereas the authors in [Hao et al., 2019] considered a specific number of known identities when performing identification attacks and it was not part of the background knowledge. Third, in our work we considered the hyper-parameters of the obfuscation technique as part of the background knowledge. Last but not least, we adapted

the three-components adversary model (i.e., goal, knowledge and capability) to the image obfuscation application domain to clearly define and demonstrate how these different components (mainly knowledge and capabilities) affect one another.

4.5.4/ EVALUATION FRAMEWORKS

Several evaluation frameworks have been proposed in the literature to evaluate obfuscation techniques in the context of images/videos. Some frameworks rely on human participants [Li et al., 2017b] whereas others rely on quantitative metrics [Dufaux et al., 2010, Nawaz et al., 2017] e.g. SSIM, recognition algorithms...

On the one hand, the authors in [Li et al., 2017b] conducted an online experiment with 271 participants to evaluate the effectiveness of different obfuscation techniques (e.g. blurring, pixelating, inpainting...) against human recognition and how they affect the viewing experience. In our study, we employ quantitative-based metrics however we can hybridize our framework by including either a human-based adversary that attempts to recognize obfuscated/restored faces or a human-based evaluation module that attempt to assess the reconstruction of the images. On the other hand, the authors in [Nawaz et al., 2017] propose a framework that evaluates the obfuscation techniques (pixelating, blurring, complete masking, cartooning) based on the privacy and utility aspects in the context of videos via quantitative-based metrics. They assess the privacy aspect by quantifying the appearance similarity between the original and the obfuscated image and assess the utility by quantifying the structural similarity. Also, the authors in [Wu et al., 2020b] propose an adversarial framework to address the privacy preservation problem regarding action recognition in videos. The framework explicitly minimizes a hybrid loss function combining both, privacy and utility aspects in order to find an optimal level of privacy (anonymization) while maintaining a good level of utility. Our framework evaluates the robustness of obfuscation techniques by (i) simulating adversaries with different background knowledge, (ii) performing attacks (*recognition* or *restoration*-based) and (iii) evaluating these attacks via structural, verification, identification and accuracy-based metrics. In [Dufaux et al., 2010], the authors proposed a framework to verify the effectiveness of obfuscation techniques (pixelating, blurring, scrambling) by conducting recognition-based attacks via the PCA [Turk et al., 1991] and LDA [Belhumeur et al., 1997] algorithms. Also, the authors in [Korshunov et al., 2013] investigated the privacy-intelligibility trade-off by proposing a framework for evaluation of privacy filters. They applied several privacy techniques to faces (e.g. blurring, pixelating and masking) with varying intensities. The accuracy of the face recognition algorithm was considered a measure of privacy (a specific person should not be identified). Whereas, the accuracy of the face detection algorithm was used as a measure of intelligibility (a face should be detected). Similar to [Dufaux et al., 2010], they applied traditional meth-

ods for face recognition such as PCA [Turk et al., 1991], LDA [Belhumeur et al., 1997] and LBP [Ahonen et al., 2006]. They concluded that an increase in the strength of privacy filters leads to an increase in privacy and a decrease in intelligibility. Similarly, the framework proposed in [Korshunov et al., 2013] evaluates the best obfuscation technique regarding the privacy-intelligibility trade-off by varying the level of privacy and comparing the accuracy of both face detection and recognition algorithms. Here and unlike [Dufaux et al., 2010, Korshunov et al., 2013], (i) we employ DL-based approaches instead of traditional approaches, (ii) we add a level of abstraction to the framework to not only evaluate but recommend the most robust obfuscation technique and (iii) we study thoroughly how the background knowledge can limit/increase the adversary's attacking capabilities (*restoration*-based or *recognition*-based attacks) and privacy breaches.

4.6/ CONCLUSION

In this chapter, we proposed a generic and scalable framework to evaluate and recommend the most robust obfuscation techniques for specific identifying/sensitive information, such as an individual's face. The framework reconstructs/recognizes obfuscated faces via DL-assisted attacks, evaluates the reconstruction/recognition via different metrics and recommends the most robust obfuscation with regard to each metric. We presented the recommendation framework by (i) proposing a 4-layered iterative process, (ii) showcasing the framework's detailed structure when applied to a facial images dataset and (iii) defining the 3-components adversary model (goal, knowledge and capabilities) to our application domain (i.e., facial features obfuscations) with three threat levels and 3 attacking capabilities. We conducted three sets of experiments on the CelebA dataset [Liu et al., 2015]. In the first experiment, we validated our approach by implementing and testing our framework on obfuscated faces. Throughout the second experiment, we demonstrated how the adversary's attacking capabilities scale with her/his knowledge and how it increased the potential risk of breaching the identities of blurred face images. Throughout the third experiment, we studied the possible privacy breaches and the attack range of an adversary against blurred face images while lacking knowledge about the obfuscation's hyper-parameters.

5

Leveraging Deep Learning-Assisted Attacks against Image Obfuscation via Federated Learning

5.1/ Scenario and problem definition

In Chapter 4, we adapted the adversary model proposed in [Do et al., 2018, Bellare et al., 1993b, Bellare et al., 1995b] to the context of facial image obfuscation[1]. We presented the adversary as a three-component model having a **goal**, **knowledge** and **capabilities** (c.f. Figure 4.3). The goal was to recover the identities of the obfuscated faces in a target dataset. The knowledge, more specifically the background knowledge, was any sort of information regarding the anonymized target dataset, which constitutes: (i) the obfuscation technique employed to anonymize the target dataset and (ii) the identities that possess obfuscated face images in the target dataset. As for the capabilities, it represented to what extent an adversary is able to achieve her/his goal by performing DL-assisted attacks such as recognition-based or restoration-based attacks. Similar to [Hao et al., 2019], we also showed that the lack of background knowledge reduces drastically the adversary's attacking capabilities and we demonstrated that simply assuming additional background knowledge leverages these capabilities, hence the privacy breaches. Although the former assumption provides different aspects of the adversary's capabilities, it is challenging to uphold in practice. Hence, the following question arises "*Can an adversary, lacking background knowledge, leverage her/his attacking capabilities and cause more privacy breaches by collaborating with other adversaries?*"

[1] Throughout this chapter, we will use the terms obfuscation and anonymization interchangeably.

Example Scenario Let us consider a target dataset containing obfuscated face images of "Bob", "Alice" and "Trudy". The same obfuscation is used throughout the entire target dataset, e.g., all the face images are obfuscated via Gaussian blur (kernel (31x31) and σ=5). Adversaries A, B and C are attacking the target dataset at once as *standalone* entities, via recognition-based attacks, in order to **recover the identities of the obfuscated faces** i.e., they all have the same goal (c.f. Figure 5.1[2]). On the one hand, the three adversaries know that "Bob", "Alice" and "Trudy" have obfuscated face images in the target dataset i.e., each adversary can mine face images of the known identities. On the other hand, adversary A has no knowledge about the obfuscation technique employed in the target dataset, adversary B knows a different obfuscation technique than the one employed (in our case, it is pixelating 4x4) and adversary C knows the exact obfuscation technique employed (in our case it is Gaussian blurring with (31x31) kernel and σ=5). Therefore, as *standalone* entities, adversary A can train an identity recognition model on clear face images whereas adversary B can train via clear/pixelated face images and adversary C can train via clear/blurred face images. The accuracy of the recognition-based attacks would differ: for instance, the attack performed by adversary A would be less accurate than the one performed by adversary C. That is mainly due to adversary A's lack of background knowledge, with regard to the obfuscation technique in this case, in comparison with adversary C.

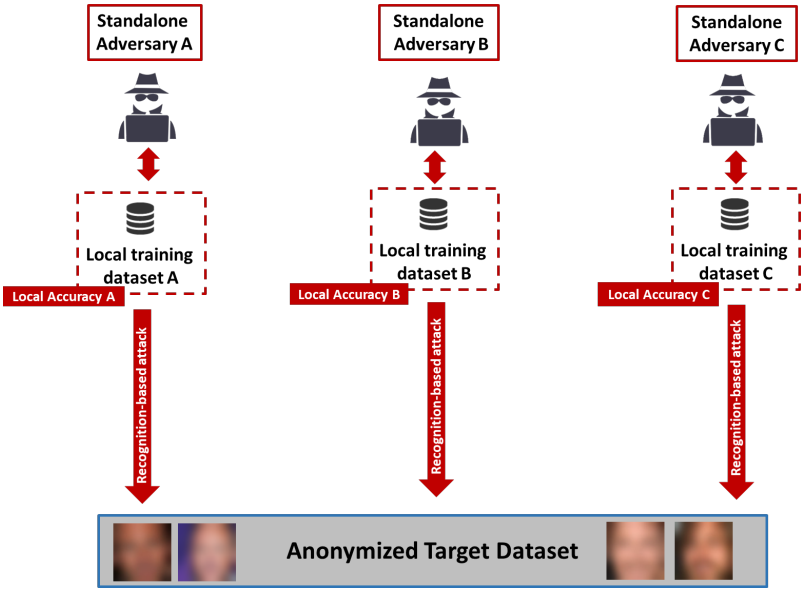


Figure 5.1: Adversaries acting as *standalone* entities and performing recognition-based attacks against the target dataset

A possible attack strategy that copes well with the lack of background knowledge would be to join forces with other adversaries and perform a collaborative attack against the target dataset. Several studies in the literature [Xu, 2008, Chen et al., 2008,

[2]It is recommended to view all the figures in this manuscript in color mode.

Duong et al., 2010] showed to what extent collaborative attacks could scale. For instance, in [Duong et al., 2010], the authors model multiple adversaries with different background knowledge and describe a mechanism for sharing the knowledge.

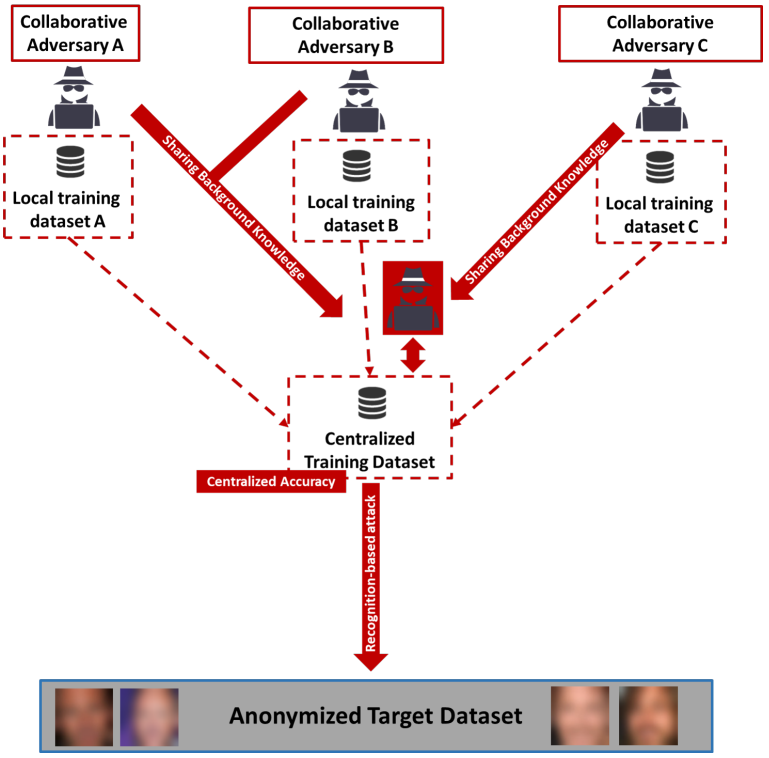


Figure 5.2: Adversaries collaborating via traditional DML by sharing their datasets with a server node while attacking the target dataset

On the one hand, when performing DL-assisted attacks, collaboration usually compels the engaged adversaries to share their local training datasets in a centralized location and delegate a server node to train a unified model via parallelism over different worker nodes i.e., traditional Distributed Machine Learning (DML) scenario (c.f. Figure 5.2) [Verbraeken et al., 2020b]. On the other hand, sharing/disclosing the local training datasets with other adversaries compromises the data privacy and depreciates its value. As stated in [Yang et al., 2019a], any sort of information, especially raw data, is most valuable when it is kept private.

Federated learning (FL) [McMahan et al., 2017, Yang et al., 2019a] has recently gained much attention as a machine learning setting where multiple clients collaborate in solving a machine learning problem under the coordination of a central server/coordinator. Each client's raw data is stored locally without being exchanged nor transferred to the central server; instead, the model's parameters are shared/aggregated and used to achieve the learning objective.

Contributions In this chapter, we empirically demonstrate that FL can be used as a collaborative attack/adversarial strategy to (i) remedy the lack of background knowledge and data shortage, (ii) leverage the attacking capabilities of each participating adversary and (iii) increase the privacy breaches without the need to share/disclose the local training datasets in a centralized location (i.e. traditional DML [Verbraeken et al., 2020b]) (c.f. Figure 5.2). In our scenario (c.f. Figure 5.3), we assume the following:

- A target dataset contains obfuscated face images where the same obfuscation is used throughout the entire target dataset.
- Multiple adversaries attack the same target dataset using DL-based techniques.
- Multiple adversaries can collaborate together, in order to train more accurate DL models for more serious attacks without disclosing their own local training datasets.
- Unlike the previous chapter and [McPherson et al., 2016, Hao et al., 2019], we assume in this case that the adversaries do not have enough training data samples to train accurate DL models and attack the target dataset. In other words, the adversaries in our study suffer from data shortage in addition to their lack of background knowledge.

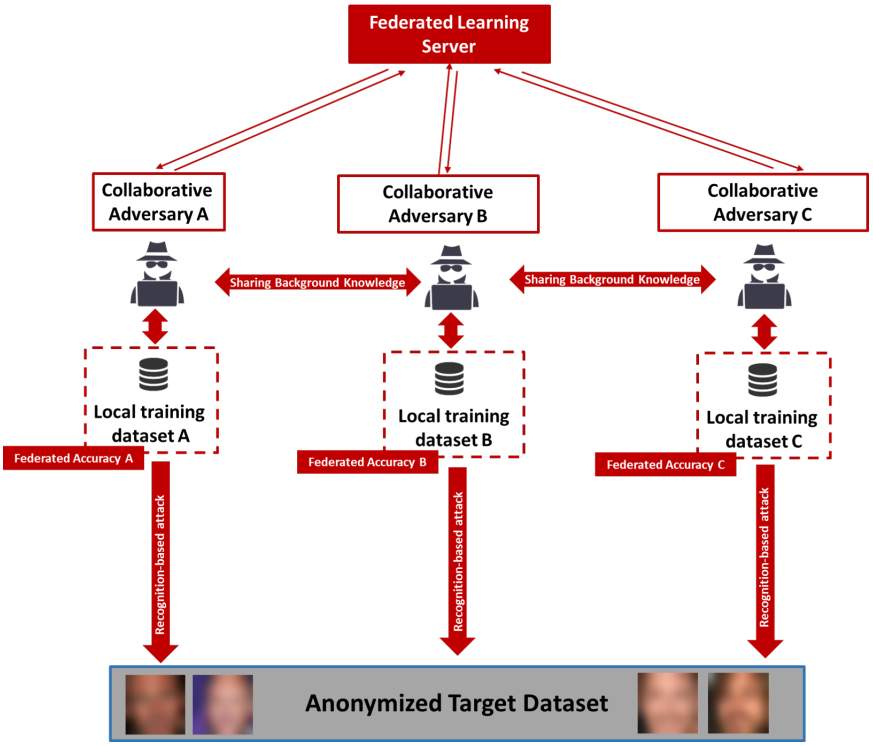


Figure 5.3: Adversaries collaborating via FL without disclosing their local datasets while attacking the target dataset

We define and study 7 **collective threat levels** based on the background knowledge of

the different adversaries and the sharing of their knowledge. We also consider *recognition*-based attacks as attacking capabilities and Gaussian blurring [Frome et al., 2009, Hill et al., 2016b] as the obfuscation technique employed in the target dataset. In addition, we focus on individuals' faces because they are the most identifying and revealing in the context of images.

The remainder of this chapter is organized as follows. In Section 5.2, we review the basic FL concepts. We present our FL-based collaborative attack in Section 5.3 and the different collective threat levels in Section 5.4. Section 5.5 evaluates the privacy breaches caused by the FL-assisted attack throughout the different threat levels. In Section 5.6, we investigate works related to collaborative attacks and the lack of background knowledge in the context of face obfuscation. This chapter is currently under peer review in a scientific journal [Tekli et al., NDa].

5.2/ PRELIMINARIES: FEDERATED LEARNING

FL is a collaborative learning technique that allows different parties (clients) to build a joint machine learning model by training locally on their datasets and sharing only the model's parameters/weights [McMahan et al., 2017, Kairouz et al., 2019]. In general, the FL system is based on a client-server architecture[3] (c.f. Figure 5.4). Let $K = \{k_0, ..., k_N\}$ denote the set of K clients, each of which has a local private dataset. The training process consists of the following 5 steps, i.e. a FL round *(step 1-5)*:

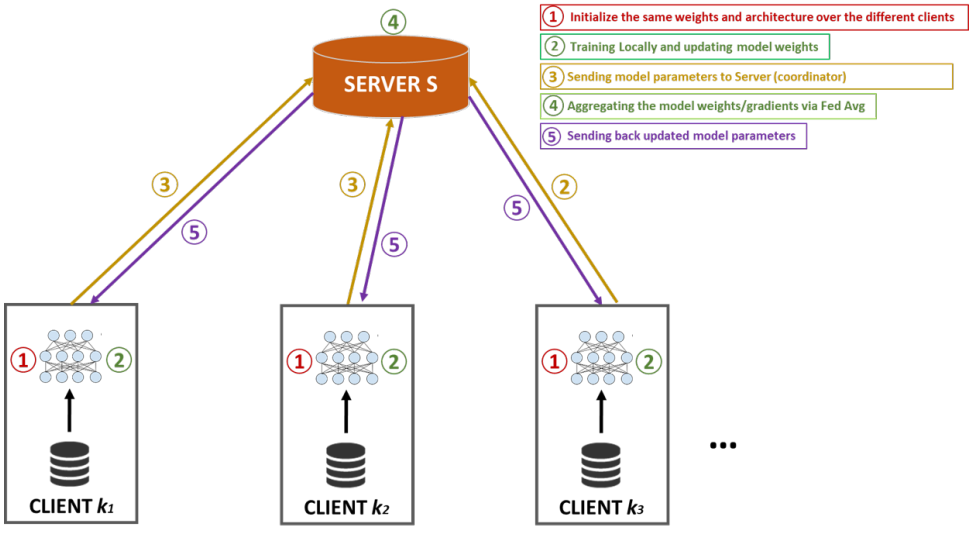


Figure 5.4: Client-Sever communication in a FL scenario

1. Server S initializes the initial parameters and forwards them to the different clients

[3] In some cases, the clients are in a peer to peer communication (no server).

participating in the current round.

2. Each client k_i loads the parameters and trains a local model on its local training dataset.
3. Each client k_i shares the local model's parameters with the server.
4. Server S aggregates the shared local models via weighted average and generates a global model.
5. Server S pushes back the new global model to the clients which update their local models respectively.

The FL process consists of numerous rounds until the global model converges. The performance of the resulting FL model should be a good approximation of an ideal model where the local datasets are grouped for a centralized training. The widely used FL algorithm for averaging the models' weights is the one proposed by McMahan in [McMahan et al., 2017] i.e., federated averaging.

The FL server can select a *C*-fraction of clients for each round and compute the weighted average over the data held by these clients. For instance, $C = 1$ means the server waits for the updates of "all" the clients participating in the FL process[4] before initiating the next round.

The authors in [Kairouz et al., 2019] categorize the FL applications as (1) cross-device and (2) cross-silo. In a cross-device FL setting, the clients represent a very large number of mobile or IoT devices ($|K|$ up to 10^{10} clients) where each client likely participates once in the FL process ($0 < C < 1$) [Hard et al., 2018, McMahan et al., 2017]. Whereas in a cross-silo FL setting, the clients are mainly different organizations (e.g. medical [Courtiol et al., 2019, Reina et al., 2019], industrial[5]) or geo-distributed datacenters ($2 < |K| < 100$ clients) where each client participates in each round of the FL process (usually $C = 1$). In our collaborative attack scenario, we consider the cross-silo setting with 3 clients (adversaries)[6] and $C = 1$ (i.e., synchronous communication where the FL server does not initiate the next round until it receives the updates from all the clients).

In addition, each client participating in the FL process is an independent entity and has complete autonomy over its local dataset [Yang et al., 2019a]. Therefore, as stated in [Wu et al., 2020a, Kulkarni et al., 2020], three main challenges arise:

[4] The communication between the FL server and the clients can be either synchronous or asynchronous. Throughout a synchronous communication, the server waits for all clients to send their update before he starts with the aggregation process.

[5] url=http://musketeer.eu/project/

[6] We can easily leverage the collaborative attack by adding additional clients (adversaries) to the FL process however we fixed the number of clients (adversaries) to 3 because it is sufficient to highlight the different aspects of the background knowledge with regard to (i) the obfuscation technique, (ii) the identities in the target dataset and show that it affects the FL process.

- **Device heterogeneity**: means that the clients have different computation, storage or communication capacities.

- **Data heterogeneity**: represents imbalanced and not identically distributed data due to the following aspects [Kairouz et al., 2019]:

  - ***Same label, different features***: the same label can have training images with different features at different clients (adversaries), e.g. due to different background knowledge with regard to the obfuscation technique. *For example, two adversaries A and B are collaborating via FL to train a recognition model and recognize the blurred face images of "Bob": both adversaries have access to clear face images of "Bob". However, adversary A knows that blurring, with a kernel size (31x31) and a standard deviation of 5, is used to obfuscate Bob's face images in the target dataset whereas adversary B has no knowledge whatsoever. Therefore, adversary A trains a recognition model on clear and blurred face images of Bob's face whereas adversary B trains a recognition model on clear face images. In other words, both adversaries will have face images of the same label "Bob" with different features (clear and blurred images with adversary "A" but only clear face images with adversary "B")*.

  - ***Label distribution skew***: when clients (adversaries) are tied to particular geo-regions or have different external knowledge (c.f. Chapter 4), the distribution of labels may vary across them. *For example, two adversaries A and B are collaborating via FL to train a recognition model and recognize the blurred faces of "Bob" and "Alice": As part of their external knowledge, Adversary A has access to Bob's social media accounts whereas Adversary B has access to Alice's accounts. Therefore, adversary A can mine images of Bob's face to her/his training dataset and adversary B can do the same with regard to Alice's face images. Therefore, each adversary possesses training images for only a single identity/label although she/he attempts to recognize both "Bob" and "Alice"*.

  - ***Quantity skew***: different clients (adversaries) can hold different quantities of training data. *For example, two adversaries A and B are collaborating via FL to train a recognition model and recognize the blurred faces of "Bob" and "Alice". Adversary A possesses 7 training examples for "Bob" and 19 for "Alice". Whereas adversary B possesses 20 training images for "Bob" and 5 for "Alice"*.

In both *quantity* and *label distribution skew* scenarios, the clients attempt to recognize the same labels i.e., the clients have the same number of classes in the final Fully Connected (FC) layer of the neural network (all the clients possess the exact neural network architecture). On the one hand, a *label distribution skew* scenario arises when a client does not possess "any" training image with regard to certain labels whereas the other

clients do. On the other hand, a *quantity skew* scenario appears when all clients possess training images for each label however in different quantities.

There are additional aspects of the *data heterogeneity*, however we listed above the ones that are the most relevant to our study [Kairouz et al., 2019].

- **Model heterogeneity**: denotes that each client needs a model specifically customized for her/his environment. In other words, each client (adversary) has a different model architecture, in our case a different neural network architecture. *For instance, two adversaries A and B are collaborating via FL to train their recognition models and attack a target dataset containing obfuscated face images of "Bob", "Alice" and "Trudy". Adversary A attempts to recognize the obfuscated images of "Bob" whereas adversary B attempts to recognize the images of "Alice" and "Trudy". Both adversaries employ the same classifier. However, adversary A's network has 1 class in the final FC layer whereas adversary B's network has 2 classes. In other words, the final FC layers differ between the two classifiers. Hence, the model heterogeneity.*

5.3/ COLLABORATIVE ADVERSARIES

In this chapter, we consider multiple adversaries[7] collaborating via FL in order to attack a target obfuscated dataset (c.f. Figure 5.3). The adversaries in our FL-based collaborative attack are characterized as follows:

1. They have the **same goal**; recover the identities of the anonymized images in the target dataset.

2. They can/might suffer, with varying degrees, from lack of background knowledge. In other words, the adversaries can/might have **different background knowledge** with regard to (i) the obfuscation technique employed in the target dataset and (ii) the identities *present*[8] in the target dataset.

3. They can/might **share their background knowledge** with one another. As stated in [Yang et al., 2019a, Xu, 2008], during the training process of FL, clients can exchange information as long as the exchange does not reveal any protected private portions of the data on each site. In this work, we consider two sharing scenarios: (1) none of the adversaries share their background knowledge (i.e., worst case sce-

[7] We consider that all adversaries are based on the three-component model discussed in the previous chapter.

[8] That possess obfuscated face images

nario) or (2) all adversaries share all their background knowledge with one another (i.e., best case scenario)[9].

4. They do not have enough training data samples to train an accurate DL-assisted classifier and perform a strong recognition-based attack as a standalone entity, i.e., they **suffer from data shortage due to small local training datasets**.

5. They perform **recognition-based attacks** against the target dataset, i.e., they train deep convolutional neural networks to perform recognition tasks on obfuscated face images such as in Chapter 4 and [McPherson et al., 2016].

6. They have **different** clear/non-obfuscated **training datasets**. For instance, even if multiple adversaries are trying to recognize the same identity, they will have different training images belonging to that particular identity.

7. They are **honest** and **exchange accurate information** with one another and with the server[10].

8. They have the **same computing capabilities** (the *device heterogeneity* discussed in Section 5.2 is not considered in this study).

5.4/ COLLECTIVE THREAT LEVELS

As mentioned previously, the background knowledge of an adversary is any information about (i) the obfuscation technique and (ii) the identities *present* in the target dataset. The threat level of an adversary depends heavily on her/his background knowledge [Hao et al., 2019] (c.f. Chapter 4). *For instance, an adversary can perform a recognition-based attack only when equipped with partial/full knowledge about the identities present in the target dataset.* In our case, because multiple adversaries are collaborating to attack a target dataset, we identify **seven collective threat levels** based on the background knowledge of the different adversaries and the sharing of this knowledge. We defined the threat levels by answering 3 main questions:

- **"Do the adversaries lack background knowledge about the obfuscation technique used to anonymize the face images in the target dataset?"**
- **"Do the adversaries lack background knowledge about the identities *present* in the target dataset?"**

[9] Additional sharing scenarios can be explored in future studies where some (not all) adversaries decide to share part of their background knowledge

[10] Issues with regard to attacking the FL setting, e.g. via model poisoning [Bagdasaryan et al., 2020] and data poisoning [Biggio et al., 2012, Liu et al., 2018], are not part of this study's scope.

Table 5.1: Collective threat levels

Questions \ Threat Levels	T_1	T_2	T_3	T_4	T_5	T_6	T_7
Do the adversaries lack knowledge about the obfuscation technique used in the target dataset?	No	Yes		No		Yes	
Do the adversaries lack knowledge about the identities *present* in the target dataset?	No	No		Yes		Yes	
Are the adversaries sharing their BK ?	NA	No	Yes	No	Yes	No	Yes

- **"Are all the adversaries sharing their background knowledge with one another?"** (c.f. Table 5.1).

Therefore, we defined the following threat levels:

- **Threat Level T_1**: assumes that none of the adversaries lack background knowledge. In other words, each adversary collaborating via FL knows (i) the exact obfuscation technique used to anonymize the target face images and (ii) all the identities *present* in the target dataset (i.e., full knowledge regarding the identities). Hence, sharing the background knowledge in this case does not have additional effect because the adversaries already possess the same information.

- **Threat Levels T_2/T_3**: assume the adversaries lack knowledge about the obfuscation technique used to anonymize the target dataset. In other words, all the adversaries (i) have full knowledge regarding the identities *present* in the target dataset however (ii) not all of them know the exact obfuscation technique employed to obfuscate the face images. Sharing the background knowledge between the adversaries may affect the FL process, the attacking capabilities and the privacy breaches which is why we talk about two threat levels $\mathbf{T_2}$ and $\mathbf{T_3}$. We consider $\mathbf{T_2}$ the collective threat level when the adversaries decide to not share their background knowledge whereas $\mathbf{T_3}$ is when the background knowledge is shared[11].

- **Threat Levels T_4/T_5**: assume the adversaries lack knowledge about the identities *present* in the target dataset. In other words, all the adversaries are aware of (i) the exact obfuscation technique used to anonymize the face images in the target dataset, however, (ii) they have partial knowledge regarding the identities. Sharing the background knowledge can have an effect on the FL process, attacking capabilities and the privacy breaches which is why we mention two threat levels $\mathbf{T_4}$ (background knowledge is not shared) and $\mathbf{T_5}$ (background knowledge is shared)[12].

[11] We study thoroughly the difference between $\mathbf{T_2}$ and $\mathbf{T_3}$ in use case 2, Section 5.5.2.
[12] Refer to use case 3 in Section 5.5.2 for more details.

- **Threat Levels T_6/T_7**: assume the adversaries lack knowledge about both the obfuscation technique and the identities *present* in the target dataset. Sharing the background knowledge can have an effect on the FL process, attacking capabilities and the privacy breaches which is why we talk about two threat levels **T_6** (background knowledge is not shared) and **T_7** (background knowledge is shared)[13].

5.5/ EXPERIMENTS

5.5.1/ EXPERIMENTAL SETUP

The main idea of our attack is to leverage the capabilities of *standalone* adversaries that lack background knowledge by jointly training deep neural networks via FL to recognize the identities of obfuscated faces (i.e., perform recognition-based attacks). In our experimental setup, we consider a cross-silo FL setting where the number of clients (adversaries) is 3 and all of them participate in the FL process (i.e. send their weights to the server) during each round (i.e., $C = 1$, c.f. Section 5.2). As mentioned in section 5.2, we can easily leverage the collaborative attack by adding additional clients (adversaries) to the FL process, however, we fixed the number to 3 because it is sufficient to highlight the different aspects of the background knowledge and show that it affects the FL process. We also assume that the 3 clients (adversaries) are employing the same classifier to perform the recognition-based attacks[14] and have access to a limited amount of clear face images per known identity[15], i.e. data shortage.

We implemented/adapted the original federated averaging algorithm proposed in [McMahan et al., 2017] based on the publicly available code[16] [Luo et al., 2019]. In all of our experiments, the training datasets and the test dataset are disjoint.

Obfuscation technique In this chapter, we consider a target dataset obfuscated via the blurring technique with a kernel size (31x31) and a standard deviation of 5. Blurring is a degradation technique utilized in image processing. It can be generated by a Gaussian kernel. It removes details from an image by applying a Gaussian kernel. The blurriness level is controlled by the standard deviation σ. Other obfuscation techniques and degrees should/will be considered in future work.

[13] Refer to use case 4 in Section 5.5.2 for more details.

[14] In a real scenario, the adversaries (clients) participating in the FL process can agree on the classifier's choice before the start of the collaborative attack. In addition, a new adversary (client) willing to join the FL process must employ the same classifier.

[15] In a real scenario, these images can be mined from social media accounts, public datasets...

[16] https://github.com/FederatedAI/FATE

Dataset We used the publicly available FaceScrub dataset for training and testing purposes[17]. The FaceScrub dataset [Ng et al., 2014] is a large dataset originally consisting of 106,863 face images of 530 female and male celebrities. For our experiments, we randomly selected a set of 100 identities denoted as N (50 female, 50 male celebrities) with 70 clear face images each. Out of these 70 images, 5 were left for testing and the remaining 65 images were distributed over the different adversaries for training purposes (i.e., our test dataset contains $|N| * 5$= 500 face images). We resized all the images to 64x64 and applied the blurring function (31x31) afterwards (c.f. Figure 5.5).

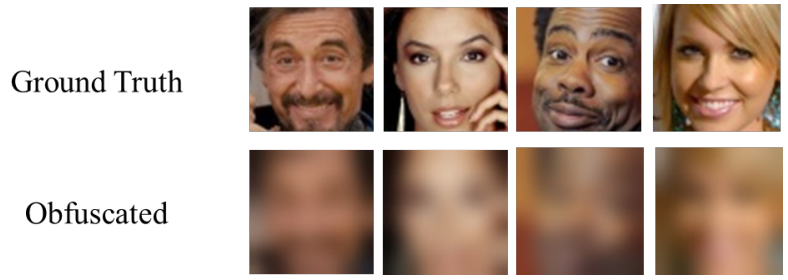


Figure 5.5: Obfuscated and Ground Truth images from the FaceScrub dataset

Three training setups We performed three different training setups for each experimental scenario: First, we considered the normal **FL** process where the adversaries collaboratively train without sharing their datasets but only their models' weights. Second, we grouped the datasets of the different adversaries and launched a **centralized** training (traditional DML setting) in order to show how good/bad approximation the FL model is, compared to the ideal centralized model. Third, we trained each adversary as a **standalone entity** similar to Chapter 4 and [Yu et al., 2020, Hao et al., 2019] in order to showcase how the FL process affects the capabilities of the standalone adversaries and the privacy breaches. We employed the same configuration and hyper-parameters for the **centralized**, **FL** and **standalone** setups for fair comparison. In addition, we used "clear" and "obfuscated" versions of each face image in the training datasets as an image augmentation process.

Training Implementation Throughout our experiments, we used the Resnet50[18] architecture as the local neural network (classifier) for each adversary [He et al., 2016c]. In addition, we adapted the federated averaging algorithm implemented in pytorch [Luo et al., 2019] to our Resnet50 classifier[19]. Similar to [Luo et al., 2019], we replaced the server-client communication with saving and restoring checkpoints on hard-devices.

[17] http://vintage.winklerbros.net/facescrub.html

[18] The implementation of the resnet50 architecture is provided by the pytorch framework via https://pytorch.org/docs/stable/_modules/torchvision/models/resnet.html#resnet50.

[19] The code will be published alongside the scientific contribution

Hence, the adversaries (clients) and the server were always deployed on the same machine (i.e. same computing capabilities and stable communication between the clients and the server).

Training parameters We specified the number of rounds executed between the adversaries and the server as 500. Each adversary executed 1 local epoch ($l = 1$) with a batch size of 4 ($b = 4$) over her/his own local training dataset before sharing the model's weights with the server. We trained the Resnet50 networks [He et al., 2016c] via RMSprop optimizer [Hinton et al., 2016, Ruder, 2016] with an initialization learning rate of 10^{-5}. Also, we initialized all the local Resnet50 networks with the ImageNet pre-trained weights [Russakovsky et al., 2015b]. We unfroze all the layers of the Resnet50 network while training, that way the back-propagation would affect not only the last FC layer but all the network's layers. As mentioned before, our code was implemented and executed using Pytorch [Paszke et al., 2019] on a machine with Intel Xeon E5-2698 CPUs and 8 Tesla V100 GPUs (cuda version 10.1).

5.5.2/ EXPERIMENTAL USE CASES

We designed and performed 4 experimental use cases (c.f. Table 5.2) in order to show how the lack of background knowledge along with data shortage can limit the adversaries' capabilities to breach the target dataset's privacy and how these limitations are remedied via FL.

Table 5.2: Use case description

Use Case	Studying the effect of:
1	Data shortage
2	Data shortage + lack of background knowledge with regard to the obfuscation technique used in the target dataset
3	Data shortage + lack of background knowledge with regard to the identities *present* in the target dataset
4	Data Shortage + lack of background knowledge with regard to obfuscation technique and the identities *present* in the target dataset

We used two metrics throughout the four experimental use cases: (1) the Top-1 recognition accuracy of the recognition-based attacks over the obfuscated test set and (2) the number of accurately recognized individuals[20] (c.f. Section 4.3.2). Also, we performed each training two times and reported the average values, with regard to both metrics, in the section below.

[20]Our test set contains 5 anonymized face images per individual. Hence, we consider that an individual is accurately recognized if L images out of 5 are recognized (Top-1 recognition) where $0 < L <= 5$. In our experiments, we report the values for L= 3.

Use case 1 (T_1) In this use case, we demonstrate that data shortage (i.e., small local training datasets) limit the capabilities of standalone adversaries and that these capabilities can be remedied and leveraged via FL. We consider threat level $\mathbf{T_1}$ where none of the adversaries lack background knowledge. In other words, each adversary trains a classifier via "clear" and "blurred" face images to recognize the blurred faces of the $|N| = 100$ identities *present* in the target dataset.

We conducted 4 scenarios by modifying (a) the number of face images per identity (label) in the local training datasets of the adversaries (scenario 1.a-1.c in Table 5.3) and (b) the number of adversaries participating in the FL-assisted attack (scenario 1.d). In short, we show that:

– FL leverages the attacking capabilities and remedies the shortage of local training datasets.

– Increasing the number of adversaries participating in the FL-assisted attack leverages the attacking capabilities and intensifies the privacy breaches

Table 5.3: Experimental set up in use case 1

Use Case 1					Dataset Distribution
Adversary		A	B	C	
Background Knowledge	Obfuscation Technique	Blurring (31,31)	Blurring (31,31)	Blurring (31,31)	
	Known identities	Set of identities N	Set of identities N	Set of identities N	
Images Features in local training datasets		Clear / Blurred images	Clear / Blurred images	Clear / Blurred images	
T_1	Scenario 1.a	9 clear images per label	9 clear images per label	9 clear images per label	identically distributed
	Scenario 1.b	14 clear images per label	14 clear images per label	14 clear images per label	identically distributed
	Scenario 1.c	19 clear images per label	19 clear images per label	19 clear images per label	identically distributed

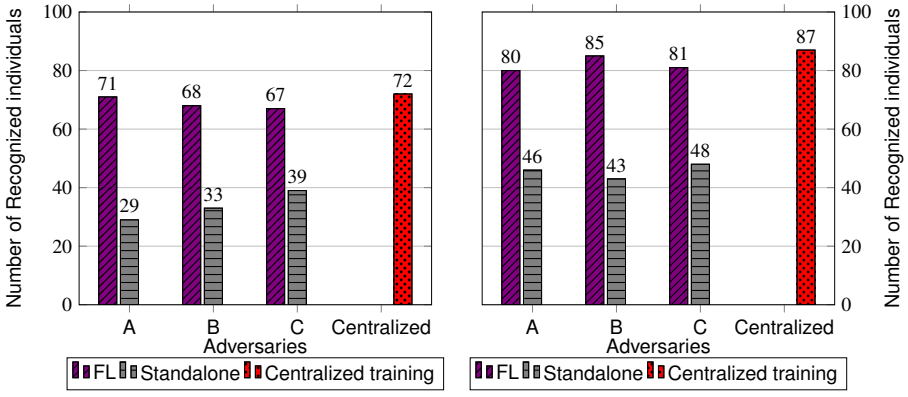
– **FL leverages the attacking capabilities and remedies the shortage in local training datasets** (*scenario 1.a, 1.b and 1.c*): In comparison with the *standalone* setups, adversaries A, B and C breach the privacy of the individuals in the target dataset by almost 50% more after collaborating via FL. For instance as visible in Figure 5.6.a and Table 5.4 (scenario 1.a), adversary A recognized 29 individuals when performing a *standalone* recognition-based attack (top-1 accuracy of 36.2%) however she/he recognized 71 individuals when collaborating with the other two adversaries through FL (61.8%)[21]. Similar behavior persists throughout scenario

[21] Basically in the pytorch framework, the weights of a network can be saved via two methods *state_dict()* or *params()*. *state_dict()* saves the weights containing both parameters and persistent buffers (e.g., Batch Normalization's running mean and var), i.e. the complete weight structure. Whereas *params()* only saves the parameters without the persistent buffers. In our implementation, when averaging we are saving and loading the parameters via *param()*. The buffers for each network will not be shared nor aggregated. Therefore, the local classifiers' performances will be close but not identical.

1.b and 1.c as visible in Figure 5.6.b-c and Rable 5.4, proving that data shortage (i.e. small local training datasets) can be remedied via FL-based collaboration.

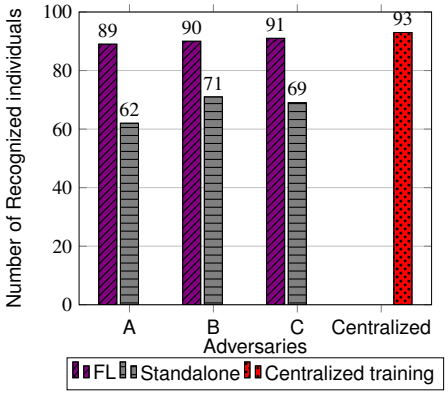
Table 5.4: Top-1 accuracy of recognition-based attacks in use case 1

Top-1 accuracy				
Adversary		**A**	**B**	**C**
Scenario 1.a	**FL**	61.8 %	61.2 %	61.6 %
	Standalone	36.2 %	39.6 %	41.4 %
	Centralized	63.2 %		
Scenario 1.b	**FL**	68.2 %	71 %	69.8 %
	Standalone	47.6 %	46 %	48.8 %
	Centralized	72.4 %		
Scenario 1.c	**FL**	75.4 %	75.6 %	77 %
	Standalone	55 %	60.6 %	59.4 %
	Centralized	77.2 %		



(a) Scenario 1.a

(b) Scenario 1.b



(c) Scenario 1.c

Figure 5.6: Number of recognized individuals by the recognition-based attacks in use case 1

– **Increasing the number of adversaries participating in the FL-assisted attack leverages the attacking capabilities and remedies the shortage of local training datasets:** As shown in Figure 5.7 and Table 5.5, we notice that the number of recognized individuals increased along with the number of adversaries participating

in the collaborative FL-assisted attack (In this scenario, we considered that each adversary has 14 ”clear” and 14 ”blurred” face images per identity). For instance, when a *standalone* adversary performed a single attack against the target dataset, she/he recognized 33 individuals out of 100 (38.6%). When 2 adversaries collaborated together, the average number of recognized individuals was 50 out of 100 (50.8%) whereas when 5 adversaries collaborated together, the average number was 89 out of 100 (71.2%).

Table 5.5: Top-1 accuracy of recognition-based attacks in scenario 1.d

Number of Adversaries in the FL-based attack	Top-1 accuracy				
1	38.6 %				
2	50.8 %	48.6 %			
3	60.8 %	60.6 %	60.6 %		
4	64.4 %	64.8 %	64.2 %	65.6 %	
5	71.2 %	71.6 %	71.2 %	71.4 %	70.8 %

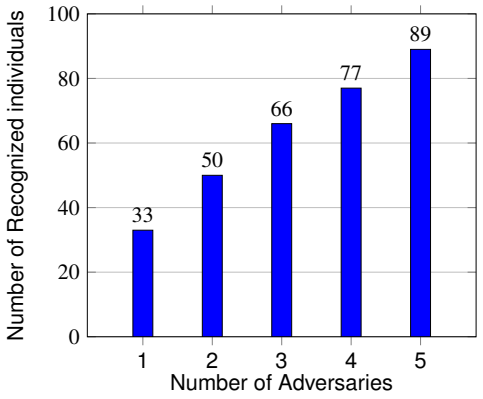


Figure 5.7: Average Number of recognized individuals by the adversaries in scenario 1.d

Use case 2 (T_2/T_3) In this use case, we demonstrate that the lack of background knowledge, with regard to the obfuscation technique used to obfuscate the face images in the target dataset, limits the attacking capabilities of *standalone* adversaries and that these limitations can be remedied via FL. In short, we demonstrate that:

- FL leverages the attacking capabilities and remedies the lack of background knowledge with regard to the obfuscation technique.
- Sharing the background knowledge, with regard to the obfuscation technique, intensifies the privacy breaches and leads to more serious FL-assisted attacks.

As seen in Table 5.6, the adversaries, and more specifically adversaries A and B, lack background knowledge about the obfuscation technique employed in the target dataset. Therefore in $\mathbf{T_2}$, each adversary trains a classifier using different images features: for

instance adversary A trains a classifier model with “clear” face images (the number of face images per identity/label in the training set is 14), whereas adversary B trains with “clear” and “pixelated” face images and adversary C trains with “clear” and “blurred” face images (the number of face images per identity/label in adversary B and C’s local training datasets doubles to 28). Therefore, we talk in $\mathbf{T_2}$ about *data heterogeneity* with regard to the “*quantity skew*” and most importantly the “*same label different features*” aspect (c.f. Section 5.2).

In $\mathbf{T_3}$, after sharing their background knowledge, the adversaries possess the same information with regard to the obfuscation technique. As a consequence, their local training datasets have the same features, i.e., a “clear”, “pixelated” and “blurred” version of each face image. Therefore, the *data heterogeneity* with regard to both “*quantity skew*” and the “*same label different features*” aspects is lifted.

Table 5.6: Experimental setup use case 2

Use Case 2					Dataset Distribution
Adversary		**A**	**B**	**C**	
Background Knowledge	**Obfuscation Technique**	Unknown	Pixelating (4x4)	Blurring (31,31)	
	Known identities	Set of identities N	Set of identities N	Set of identities N	
T_2	**Sharing of BK**	No	No	No	*Data heterogeneity* in terms of *“Same labels, different features”* and *“Quantity skew”*
	Images Features	Clear images	Clear / Pixelated images	Clear / Blurred images	
	Number clear Images per label	14	14	14	
T_3	**Sharing of BK**	Yes	Yes	Yes	*Identically distributed*
	Images Features	Clear / Pixelated / Blurred images	Clear / Pixelated / Blurred images	Clear / Pixelated / Blurred images	
	Number clear Images per label	14	14	14	

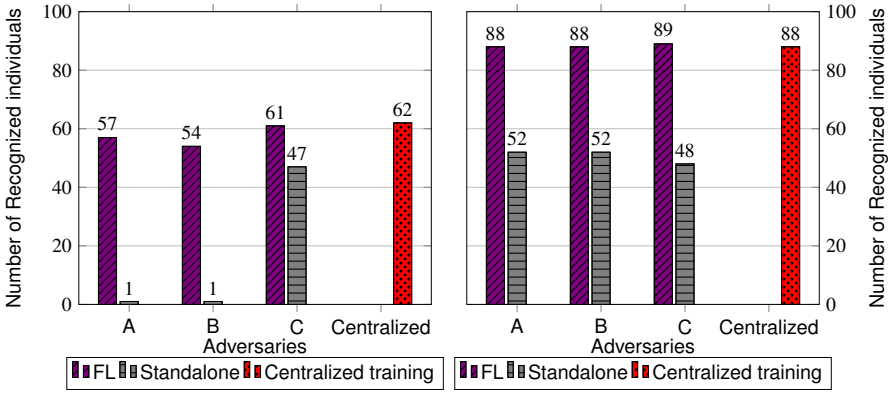
– **FL leverages the attacking capabilities and remedies the lack of background knowledge with regard to the obfuscation technique**: On the one hand, we notice in $\mathbf{T_2}$ (c.f. Figure 5.8 and Table 5.7) that both adversaries A and B, when performing as *standalone* entities, recognized only 1 individual in the target dataset (top-1 accuracy of 1.4% and 1.3%). That is because the face images in their local training datasets are “clear” and “pixelated” whereas they are trying to attack “blurred” face images. After collaborating with adversary C via FL, both adversaries recognized respectively 57 (52.6%) and 54 (52.2%) individuals in the target dataset, i.e., roughly a 52% increase in privacy breaches. In other words, adversary A and B, with no knowledge about the exact obfuscation technique in the target dataset, breached the privacy of roughly 52% of the individuals in the target dataset by only sharing their model weights with adversary C. Besides, the accuracy of the recognition-based attacks performed by adversary C was almost the same. On the

other hand, after sharing their background knowledge in **T_3**, we notice that the adversaries A, B and C recovered the identities of almost 50 individuals in the target dataset (51% as top-1 accuracy) when performing as *standalone* entities (because they all trained their models with “clear”, “pixelated” and “blurred” face images). In addition after collaborating via FL, the three adversaries breached the privacy of the individuals in the target dataset almost 20% more. On average after collaborating via FL, the adversaries recognized the identities of 88 individuals out of 100 (72.4%).

– **Sharing the background knowledge, with regard to obfuscation technique, intensifies the privacy breaches and leads to more serious FL-assisted attacks**: When comparing the two threat levels **T_2** and **T_3**, we notice that sharing the background knowledge between the adversaries (with regard to the obfuscation techniques in this case) resulted in more serious FL-assisted attacks. As visible in Table 5.7 and Figure 5.8, when collaborating via FL, the adversaries in **T_2** were able to recognize the identities of 58 individuals (53%) on average, whereas in **T_3**, they recognized 88 individuals (72%).

Table 5.7: Top-1 accuracies of recognition-based attacks in use case 2

Use Case 2				
Adversary		A	B	C
T_2	**FL**	52.6 %	52.2 %	55 %
	Standalone	1.4 %	1.3 %	48.6 %
	Centralized	58.5 %		
T_3	**FL**	72.9 %	73.2 %	73 %
	Standalone	51.5 %	52.9 %	52.1 %
	Centralized	74.9 %		



(a) Threat level T_2 (b) Threat level T_3

Figure 5.8: Number of recognized individuals by recognition-based attacks in use case 2

Use case 3 (T_4/T_5) In this use case, we demonstrate that the lack of background knowledge with regard to the identities *present* in the target dataset, can limit the attacking capabilities of standalone adversaries and that these capabilities can be remedied via FL-based collaboration. In short, we show that:

– FL leverages the attacking capabilities of the adversaries and remedies their lack of background knowledge with regard to the identities *present* in the target dataset, only when the adversaries share their background knowledge.

As seen in Table 7, the three adversaries A, B and C know that blurring is the obfuscation technique employed in the target dataset (i.e. they all have "blurred" and "clear" versions of each face image in their local training datasets). However, we consider that adversary A knows a set of identities D_A, adversary B is aware of a set of known identities D_B and adversary C of a set D_C where $D_A \cup D_B \cup D_C = N$. We also assume that $D_A \cap D_B \cap D_C = \emptyset$ with $|D_A| = |D_B| = 33$ and $|D_C| = 34$[22]. In other words, each adversary knows a distinct subset of the identities present in the target dataset.

Table 5.8: Experimental setup in use case 3

Use Case 3					Dataset Distribution
Adversary		**A**	**B**	**C**	
Background Knowledge	**Obfuscation Technique**	Blurring (31,31)	Blurring (31,31)	Blurring (31,31)	
	Known identities	Set of identities D_A	Set of identities D_B	Set of identities D_C	
T_4	**Sharing of BK**	No	No	No	
	Images Features	Clear / Blurred images	Clear / Blurred images	Clear / Blurred images	
	Number clear Images per label	14	14	14	*Model heterogeneity*
T_5	**Sharing of BK**	Yes	Yes	Yes	
	Images Features	Clear / Blurred images	Clear / Blurred images	Clear / Blurred images	
	Number clear Images per label	14	14	14	*Label distribution skew*

In **T_4**, each adversary has a distinct neural network architecture specifically with regard to the final FC layer: for instance, adversary A has the identities in D_A as classes for her/his classifier (i.e. 33 classes), whereas adversary B has D_B (i.e., 33 classes) and adversary C has D_C (i.e., 34 classes). We talk in **T_4** about *model heterogeneity* (c.f. Section 5.2). Therefore, we cannot implement the *standard* FL process where the adversaries share/aggregate the weights of the complete neural network. Instead, we implement *personalized* FL similar to [Arivazhagan et al., 2019, Yu et al., 2020] where we split the

[22] We intentionally chose $|D_C|$ to be different than $|D_A|$ and $|D_B|$ in order to highlight the model heterogeneity aspect when the adversaries do not share their knowledge and therefore differentiate in terms of neural network architecture (e.g. Adversary C has 34 output classes in the final FC layer whereas adversary A and B have 33).

neural network's layers in two parts: (a) global and (b) personalized layers. We consider the classifier layer (final FC layer) as personalized layer. Basically only the global layers' parameters are shared and aggregated by the server. After aggregation each adversary trains the personalized layers of the neural network on her/his local dataset.

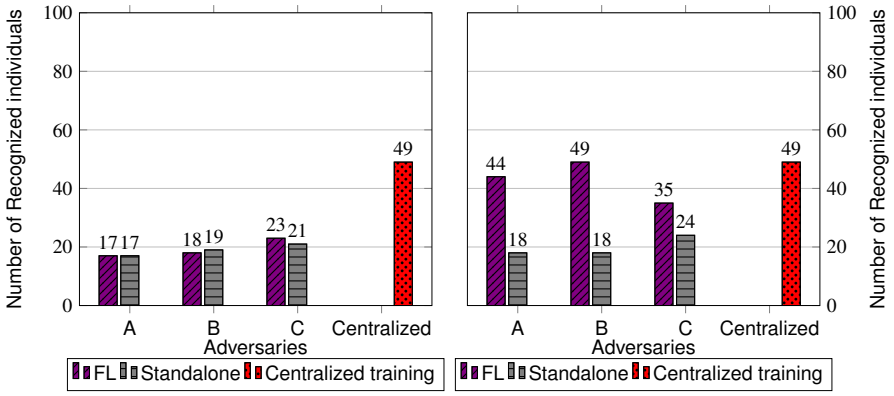
After sharing their background knowledge in **T5**, the adversaries possess the same information with regard to the identities *present* in the target dataset therefore their respective neural network architectures have the same 100 classes at the FC layer level. In other words, the *model heterogeneity* is lifted however the *data heterogeneity* with regard to the "*label distribution skew*" aspect arises (c.f. Section 5.2): for instance, adversary A has face images for only the individuals in D_A, adversary B has face images for the individuals in D_B and adversary C for D_C[23]. Therefore, we do not need to perform *personalized* FL. Our results show that:

- **FL leverages the attacking capabilities of the adversaries and remedies the lack of background knowledge with regard to the identities *present* in the target dataset, only when the adversaries share their knowledge:** We notice in **T4** (c.f. Table 5.9 and Figure 5.9.a) that the three adversaries perform similarly as *standalone* entities and after collaborating via *personalized* FL (similar behavior was observed/studied in [Yu et al., 2020]). In other words, aggregating only the global layers' weights did not affect the learning of the local classifiers. Whether the adversaries decide to collaborate or not while suffering from *model heterogeneity*, the privacy breach is roughly the same. For instance, we notice in Figure 5.9.a that adversary A recognized exactly 19 individuals when performing as a *standalone* entity and when collaborating with the other adversaries via *personalized* FL. Employing *personalized* FL in our scenario did not affect the accuracy of the recognition-based attacks, i.e., the privacy breaches did not increase after the FL-based collaboration. After sharing their background knowledge, we notice in **T5** that the adversaries are more dangerous after collaborating together via *standard* FL. For instance, adversary A recognized 17 individuals as a *standalone* entity whereas she/he was able to recognize 44 after participating in the FL-assisted attack although she/he only had face images of $|D_A| = 33$ known identities (c.f. Figure 5.9.b).

[23] In this study, we consider that the adversary does not migrate face images of the newly known identities shared by other adversaries (e.g. via external knowledge) in order to study the effect of the label distribution skew on the FL-based attack.

Table 5.9: Top-1 accuracy of recognition-based attacks in use case 3

Use Case 3				
Adversary		A	B	C
T_4	FL	16.4 %	17.9 %	20.2 %
	Standalone	16.3 %	17.9 %	19.8 %
	Centralized	46 %		
T_5	FL	46 %	47 %	41.8 %
	Standalone	16.7 %	18.4 %	20.1 %
	Centralized	46 %		



(a) Threat level T_4 (b) Threat level T_5

Figure 5.9: Number of recognized individuals by recognition-based attacks in use case 3

Use case 4 (T_6/T_7) In this experimental use case, we demonstrate that the lack of background knowledge, with regard to both the obfuscation technique and the identities *present* in the target dataset, limits the attacking capabilities of *standalone* adversaries and that FL can remedy these limitations. In short, we demonstrate that:

– FL leverages the attacking capabilities of the adversaries and remedies the lack of knowledge with regard to both the obfuscation technique and the identities *present* in the target dataset, only when the adversaries share their knowledge.

As seen in Table 5.10, the three adversaries lack background knowledge.

In $\mathbf{T_6}$, each adversary has a distinct neural network architecture specifically regarding the final FC layer and trains a classifier using different images features. We talk in $\mathbf{T_6}$ about *model heterogeneity* and *data heterogeneity* with regard to "*quantity skew*" and "*same label different features*" aspects (c.f. Section 5.2). Therefore similar to $\mathbf{T_4}$ in use case 3, we implemented *personalized* FL. We also consider the classifier layer (final FC layer) as personalized layer.

Table 5.10: Experimental setup in use case 4

Use Case 4					Dataset Distribution
Adversary		**A**	**B**	**C**	
Background Knowledge	**Obfuscation Technique**	Unknown	Pixelating (4x4)	Blurring (31,31)	
	Known identities	Set of identities D_A	Set of identities D_B	Set of identities D_C	
T_6	**Sharing of BK**	No	No	No	*Model heterogeneity* and *data heterogeneity* in terms of "*different features*" and *"quantity skew"*
	Images Features	Clear images	Clear / Pixelated images	Clear / Blurred images	
	Number clear Images per label	14	14	14	
T_7	**Sharing of BK**	Yes	Yes	Yes	*Data heterogeneity* in terms of *"Label distribution skew"*
	Images Features	Clear / Blurred images	Clear / Blurred images	Clear / Blurred images	
	Number clear Images per label	14	14	14	

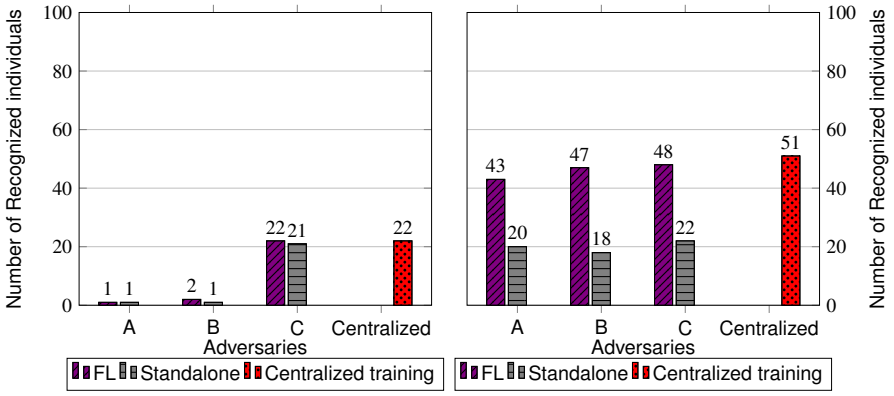
In **T**$_7$, after sharing their background knowledge, all the adversaries possess the same information with regard to the obfuscation technique and the identities *present* in the target dataset. Hence, their respective neural network architectures will have the same 100 classes at the FC layer level and their training datasets will have the same features: for instance, all adversaries will have a "clear", "pixelated" and "blurred" version of each face image in their local training datasets. In other words, the *model heterogeneity* and the *data heterogeneity* with regard to "*quantity skew*" and "*same label different features*" aspects are lifted. Therefore, we do not need to perform *personalized* FL. However, the *data heterogeneity* with regard to the "*label distribution skew*" aspect arises (i.e., adversary A has face images for only the identities in D_A, adversary B has face images for the identities in D_B and adversary C for D_C). Our results demonstrate that:

– **FL leverages the attacking capabilities of the adversaries and remedies the lack of knowledge, with regard to both the obfuscation technique and the identities *present* in the target dataset, only when the adversaries share their knowledge** (c.f. Figure 5.10 and Table 5.11): We notice that all adversaries perform similarly as *standalone* entities and after collaborating together via *personalized* FL. Whether the adversaries decide to collaborate or not while not sharing their background knowledge, the privacy breach is roughly the same. For instance in Figure 5.10.a, adversary A recognized exactly 1 individual when performing as a *standalone* entity and when collaborating with the other adversaries via *personalized* FL. After sharing their background knowledge, we notice in **T**$_7$ that the adversaries are more dangerous when collaborating via standard FL. For instance in Figure 5.10.b, adversary A recognized 20 individuals as a standalone entity whereas she/he was able to recognize 43 after participating in the FL-assisted attack. Again, sharing the background knowledge between the adversaries led to more privacy

breaches of the target dataset.

Table 5.11: Top-1 accuracy of recognition-based attacks in use case 4

Use Case 4				
Adversary		A	B	C
T_6	**FL**	2.1 %	3.8 %	19.8 %
	Standalone	1 %	1.8 %	19.7 %
	Centralized	20.3 %		
T_7	**FL**	47.3 %	47.3 %	43.3 %
	Standalone	17.4 %	18.4 %	19.6 %
	Centralized	51.3		



(a) Threat level T_6 (b) Threat level T_7

Figure 5.10: Number of recognized individuals by recognition-based attacks in use case 4

5.5.3/ DISCUSSION

First, we compare $\mathbf{T_1}$ (more specifically scenario 1.b in use case 1) and $\mathbf{T_5}$ (use case 3) because the only difference between these two scenarios is the "*label distribution skew*" (c.f. Table 5.12). In both scenarios, the adversaries have the same number of "clear" face images per identity (e.g. 14), the same images features (e.g. "clear" and "blurred" face images) and are attacking the same identities. However, the data distribution over the different adversaries in $\mathbf{T_5}$ is unbalanced and non-identical whereas it is the opposite in $\mathbf{T_1}$. Hence, a direct comparison between these two scenarios shows the effect of the *data heterogeneity* in terms of "*label distribution skew*" on the FL process. As seen in Table 5.12, we notice a 25% decrease in terms of top-1 accuracy between threat level $\mathbf{T_1}$ (more specifically scenario 1.b) and $\mathbf{T_5}$. Similar behavior has been observed and studied with the federated averaging algorithm in [Zhao et al., 2018, Hsu et al., 2020, Hsu et al., 2019, Hsieh et al., 2020]. Possible approaches to limit the decline in accuracy would be either

Table 5.12: Threat levels comparison

Use Cases	1	2		3		4	
Threat Levels	T_1	T_2	T_3	T_4	T_5	T_6	T_7
Adversaries Dataset Distribution	*Identically Distributed*	*Data heterogeneity* in terms of "*same labels different features*" And "*quantity skew*"	*Identically Distributed*	*Model heterogeneity*	*Data heterogeneity* in terms of "*label distribution skew*" aspect	*Model heterogeneity* *Data heterogeneity* in terms of *"different features"* and *"quantity skew"*	*Data heterogeneity* in terms of *"label distribution skew"*
Average Top-1 accuracy of adversaries participating in FL	69.66 %	53.2 %	72 %	18.2 %	45.6 %	8.56 %	45.9 %
Average number of recognized individuals by the adversaries participating in FL	81	53	87	19	44	7	43

(i) to try federated averaging with server momentum [Hsu et al., 2019, Hsu et al., 2020], (ii) share a small dataset between the clients [Zhao et al., 2018] or (iii) employ group normalization instead of batch normalization [Hsieh et al., 2020].

Another observation is regarding threat level $\mathbf{T_1}$ (scenario 1.b in use case 1) and $\mathbf{T_3}$ (use case 2). Although, the simplest (almost non-realistic) setting in terms of background knowledge was the one presented in $\mathbf{T_1}$, the results reported in $\mathbf{T_3}$ were slightly better (c.f. Table 5.12). That is because in $\mathbf{T_3}$, we used three versions of each face image in the training sets ("clear", "blurred" and "pixelated") instead of only two ("clear" and "blurred") which served as an image augmentation process for the local trainings.

Also as seen in Table 5.12, the weakest FL-assisted attack was in $\mathbf{T_6}$ (use case 4) where the adversaries suffered from *model heterogeneity*, *data heterogeneity* in terms of "*same labels different features*" and "*quantity skew*". That is because they lacked background knowledge with regard to the obfuscation technique, the identities *present* in the target dataset and they did not share their knowledge with one another.

In short, we demonstrate via the above experiments three main behaviors. Throughout the 4 use cases, we demonstrate that FL leverages the capabilities of each adversary collaborating in the attack in spite of (i) the data shortage and (ii) the lack of their background knowledge with regard to both the obfuscation technique and the identities *present* in the target dataset. In use case 1, we demonstrate that increasing the number of adversaries participating in the FL-assisted attack leverages the attacking capabilities and intensifies the privacy breaches. Last but not least, throughout use cases 2, 3 and 4, we demonstrate that sharing the background knowledge between the adversaries, with regard to

both the obfuscation techniques and identities *present* in the target dataset, leverages significantly their capabilities to breach the target dataset's privacy.

5.6/ RELATED WORKS

In this section, we investigate works related to (i) the effect of the background knowledge on the adversaries' capabilities in the context of face obfuscation and (ii) to the collaborative attacks.

5.6.1/ LACK OF BACKGROUND KNOWLEDGE

We (c.f. Chapter 4) and the authors in [Hao et al., 2019] demonstrate how the adversary's capabilities to breach privacy heavily depend on the background knowledge in the context of face obfuscation. On the one hand, the authors in [Hao et al., 2019] showed that a lack of background knowledge, with regard to the obfuscation technique used to anonymize the target dataset, reduces drastically the privacy breaches: *when attacking a dataset of blurred faces, an adversary, with no idea whatsoever about the obfuscation technique, performed an identity recognition DL-assisted attack with a 0.009 accuracy*. In Chapter 4, we studied the lack of background knowledge with regard to the identities *present* in the target dataset and its effect on the privacy breaches: *for instance, when attacking a dataset of blurred face images, an adversary, equipped with 12% of the identities present in the target dataset, performed an identity recognition DL-assisted attack that re-identified almost 10% of the anonymized face images*. On the other hand, we (c.f. Chapter 4) and the authors in [Hao et al., 2019] leveraged the adversary's capabilities and increased the privacy breaches by simply considering additional background knowledge. *For instance, the authors in [Hao et al., 2019] showed that when the adversary knew the obfuscation technique along with its hyper parameters used in the target dataset, she/he performed an identity recognition DL-assisted attack with a 0.783 accuracy. Furthermore we showed in the previous chapter that when equipped with all the identities present in the target dataset, an adversary executed a recognition-based attack and re-identified almost 81% of the anonymized face images.* In addition, the authors always tend to assume that the adversaries have enough training data samples to train the DL models and attack the target dataset i.e., the adversaries never suffer from *data shortage* (c.f. Chapter 4 [McPherson et al., 2016, Hao et al., 2019]. Although the former assumptions provide different aspects of the adversary's capabilities, they are challenging to uphold in practice. Therefore, in this chapter we considered adversaries lacking background knowledge, suffering from data shortage and performing DL-assisted attacks.

5.6.2/ COLLABORATIVE ATTACKS

In [Xu, 2008], the authors envisioned the idea of collaborative attacks in the context of cyber-security. They categorized the collaborative attacks in terms of (i) time-aspect (i.e. offline/on-line coordination between the adversaries during the attack), (ii) space-aspect (i.e. centralized, distributed, peer to peer architecture. . .), (iii) information exchanged during attack (i.e. one/two-way) and (iv) effect of attack (e.g. spatial collaboration[24]). Based on their categorization, our adversaries (i) communicate with the FL server to share and average their DL models' weights before performing the attack, i.e., offline coordination, (ii) are distributed following a client-server architecture, (iii) exchange information with the server, but not with each other, in a two-way manner and (iv) perform the attack at the same time after training their local DL model via FL.

Also in [Chen et al., 2008], the authors studied malicious adversaries that perform inference attacks against a database to extract identifying/sensitve information. Similar to our study, they consider multiple adversaries working together, merging their knowledge and jointly inferring sensitive information. Also they showed that generalizing from a single-adversary to a multi-adversary collaborative system increases the information breach. Last but not least, the authors in [Duong et al., 2010] examine a network of adversaries who seek to discover the sensitive information of target individuals in a dataset. They model multiple adversaries with different background knowledge and they describe a mechanism for sharing this knowledge.

On the one hand, similar to the above studies, we consider multiple adversaries attacking a target anonymized dataset while sharing their background knowledge prior to the attack. Also, we demonstrate that the collaboration leverages the capabilities of the *standalone* adversaries and increases the privacy breaches in most of the cases. On the other hand, the main differences rely in (i) attacking a dataset of obfuscated facial images via (ii) DL-assisted privacy attacks and (iii) collaborating via FL instead of the traditional machine learning approach where all the local datasets are grouped for a centralized training. To the best of our knowledge, our work is the first that considers FL-based collaborative attacks against an anonymized images dataset.

5.7/ CONCLUSION

In this chapter, we empirically demonstrated that FL can be used as a collaborative attack/adversarial strategy to (i) remedy the lack of background knowledge and data shortage, (ii) leverage the attacking capabilities of an adversary and (iii) increase the privacy

[24]The set of adversarial computers, which are located in different geographic or network places, are coordinated to launch attacks against a target at (roughly) the same time.

breaches without the need to share/disclose the local training datasets in a centralized location. We defined seven collective threat levels based on the background knowledge of the different adversaries and the sharing of that knowledge. We conducted four experimental use cases on the Face Scrub dataset [Ng et al., 2014]. Throughout the four use cases, we showed that FL leverages the capabilities of each adversary participating in the FL-assisted attack despite data shortage and the lack of background knowledge. For instance, in one threat level, an adversary, with no knowledge about the obfuscation technique, leveraged the recognition-based attack against blurred face images and re-identified 57 out of 100 individuals instead of just 1 while only sharing the model's parameters. Second, we demonstrated that increasing the number of adversaries participating in the FL-assisted attack leads to more serious attacks and intensifies the privacy breaches. For instance, when 2 adversaries collaborated together, the average number of recognized individuals was 50 out of 100 (50.8%) whereas when 5 adversaries collaborated together, the average number was 88 out of 100 (71.2%). Last but not least, we showed that sharing the background knowledge between the adversaries increases significantly the attacking capabilities. For instance, the adversaries were able to recognize the identities of 58 individuals (53%) on average without sharing their knowledge, whereas they recognized 88 individuals (72%) when sharing it.

6

(k,l)-CLUSTERING FOR TRANSACTIONAL DATA STREAMS ANONYMIZATION

6.1/ SCENARIO AND PROBLEM DEFINITION

In Chapter 2 and more specifically Section 2.4, we discussed privacy preservation in the context of static relational datasets. We discussed (i) different privacy models proposed to protect the individual's tuples against identity and attribute disclosure by applying generalization and bucketizaton-based mechanisms, (ii) the correlation problem that arises in the context of static transactional datasets, specifically when applying anatomy [Xiao et al., 2006a], and (iii) the techniques proposed to counter act against this problem such as Safe Grouping [al Bouna et al., 2013, al Bouna et al., 2015a] and (k, l)-diversity [Gong et al., 2017]. While these techniques are useful in dealing with the correlation problem on bulk datasets, they provide no proof of effectiveness in anonymizing a data stream. These techniques assume all data to be available at initial time whereas in the context of a data stream it is quite the opposite: New tuples are generated at each instance and must be protected on the fly before being stored in an anonymized dataset. We consider that the anonymization technique has a partial view of the data stream, limited to the batch of tuples undergoing the anonymization.

Example Scenario Let us consider a car rental example scenario depicted in Figure 6.1 where each smart vehicle triggers an event between two piers in the form of a transaction to be stored in a dataset for analysis. Transactions are generated continuously as long as customers are driving their vehicles to form a data stream. In this scenario, we assume that the anonymization must be performed on the stream of tuples generated by the data source to output an anonymized dataset in the form shown in Figure 6.2.

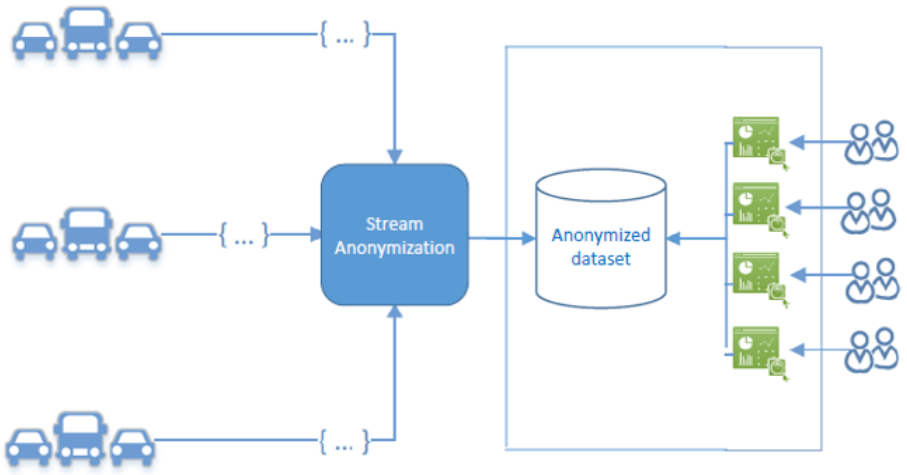


Figure 6.1: Smart car rental scenario

The same correlation problem noted in Chapter 2 (c.f. Section 2.4) arise in the context of a data stream (i.e. inter and intra QI-group correlations). *For instance, the released 2-diverse dataset is divided into two separate tables to hide the link between the identifying and sensitive values as in [Xiao et al., 2006a, Li et al., 2012, Wang et al., 2010]. In a QI-group an identifying value cannot be associated with a sensitive value with a probability greater than 1/2. The problem arises when the identifying and sensitive values correlate across the QI-groups [Amiri et al., 2018, al Bouna et al., 2013, Gong et al., 2017] (e.g. inter-group correlations in the first two QI-groups in Figure 6.2(b)). This leads to an implication that the sensitive values belong to the same individual.*

Contributions In this chapter, we extend the work in [al Bouna et al., 2013, al Bouna et al., 2015a] to address the correlation problem in the anonymization of a transactional data stream where new data is continuously generated and its distribution is imbalanced. We propose (k, l)-clustering that continuously groups k distinct individuals into l-diverse QI-groups and ensures that these individuals remain grouped together in future releases of QI-groups. (k, l)-clustering keeps track of incoming identifying values to safely release them across the QI-groups. It is a bucketization technique that prevents attribute disclosure, releasing trustful information. Our main contributions in this chapter include:

- Defining privacy properties that are required to bind the correlations in a data stream.
- Proposing a novel clustering approach to enforce the aforementioned privacy properties.

The remainder of this chapter is organized as follows. In Section 6.2, we define the basic

User ID	VIN Number	TS	Location
Allen_U1	abfb32fd10ad2*	1	10,31;17,32
Betty_U2	00983503e35d*	2	30,29;24,12
Cathy_U3	0f550377353d*	3	23,45;11,23
Allen_U1	abfb32fd10ad2*	4	10.32;15,32
David_U4	936e2c77b9du*	5	22,25;11,33
Allen_U1	abfb32fd10ad2*	6	13.32;15,32
Betty_U2	00983503e35d*	7	42,45;11,23
Cathy_U3	0f550377353d*	8	23,45;11,24
Carol_U5	1qwfq2fd10ad2*	9	40,42;14,31
Cathy_U3	0f550377353d*	10	30,30;24,12
...	...	...	...

(a) Incoming Stream S

User ID	VIN Number	GID
Allen_U1	abfb32fd10ad2*	1
Betty_U2	00983503e35d*	1
Cathy_U3	0f550377353d*	2
Allen_U1	abfb32fd10ad2*	2
David_U4	936e2c77b9du*	3
Allen_U1	abfb32fd10ad2*	3
Betty_U2	00983503e35d*	4
Cathy_U3	0f550377353d*	4
Carol_U5	1qwfq2fd10a*	5
Cathy_U3	0f550377353d*	5
...	...	...

GID	Location	TS
1	10.31;17,32	1
1	30,29;24,12	2
2	23,45;11,23	3
2	10.32;15,32	4
3	22,25;11,33	5
3	13.32;15,32	6
4	42,45;11,23	7
4	23,45;11,24	8
5	40,42;14,31	9
5	30,30;24,12	10
...	...	...

(b) Anonymized Stream S*

Figure 6.2: Rental data stream anonymized

concepts and definitions. We present our privacy model in Section 6.3 and describe the (k, l)-clustering approaches. Section 6.4 evaluates the performance of our algorithm by adopting two clustering techniques to a data stream. In section 6.5, we investigate works related to the anonymization of a data stream. This chapter was published in the International Conference on Information Security Practice and Experience ISPEC 2018 [Tekli et al., 2018].

6.2/ PRELIMINARY DEFINITIONS

In this section, we present the basic concepts and definitions to be used in the remainder of this chapter.

Definition 1: Tuple - *t*

A tuple t is a finite ordered list of values $\{v_1,v_2,...,v_b\}$ where, given a set of attributes $\{A_1,...,A_b\}$, $\forall i\ (1\leq i\leq b)\ v_i=t[A_i]$ refers to the value of attribute A_i in t. We categorize attributes as follows:

- *Identifier* (A^{id}) is an attribute whose value is linked to an individual in a given dataset. For example, a social security number anonymized in a way to represent uniquely an individual but cannot explicitly identify her/him.
- *Sensitive attribute* (A^s) reveals critical and sensitive information about a certain individual and must not be directly linked to individuals' identifying values in data sharing, publishing or releasing scenarios.
- *Time-stamp* (A^{ts}) indicates the arrival time of the tuple, its position in *S*. The time-stamp is considered identifying, which can be used to expose individuals' privacy in a transactional data stream. Here, we do not publish the time-stamp, we use it instead for evaluating the utility of our anonymization technique.

Definition 2: Data Stream - S

A *data stream* $S= t_1,t_2...,$ is a continuously growing dataset composed of infinite series of tuples received at each instance. Let *U* be the set of individuals of a specific population, $\forall\ u \in U$ we denote by S_u the set of tuples in *S* related to the individual u, where $\forall\ t \in S_u$, $t[A^{id}]=v_{id}$.

Definition 3: Cluster - C

Let $S' \subset S$ be a set of tuples in S. A cluster C over S' is defined as a set of tuples $\{t_1, ..., t_n\}$ and a centroid V_{id} consisting of a set of identifying values such that, $\forall t \in C, t[A^{id}] \in V_{id}$. We use the notation $V_{id}(C)$ to denote the centroid V_{id} of C.

Definition 4: Equivalence class / QI-group

[Samarati, 2001] A quasi-identifier group (QI-group) is defined as a subset $QI_j, j = 1, 2, ...$ of released tuples in $S^* = \bigcup_{j=1}^{\infty} QI_j$ such that, for any $j_1 \neq j_2$, $QI_{j1} \cap QI_{j2} = \emptyset$.

We stick with the QI-group terminology for compatibility with the broader anonymization literature, which can include identifying as well as quasi-identifying attributes.

Table 6.1: Notations

S	Incoming Stream
t_p	Tuple in S arriving at instance p
u	Individual described in S
S_u	Set of tuples related to individual u
A	Attribute of S
A^{id}	Identifying attribute of S
A^{sv}	Sensitive attribute of S
A^{ts}	Time-stamp attribute of S
v_{id}	Identifying value of a tuple in S
v_s	Sensitive value of a tuple in S
QI	Quasi-identifier group
$\|U\|$	Number of distinct individuals in S
$\|S\|$	Total number of tuples in S
C	Cluster over S
$V_{id}(C)$	Centroid of a cluster C
S^*	Anonymized version of S

6.3/ PRIVACY PRESERVATION

We work under the assumption that the anonymization of the data stream will continuously release l-diverse QI-groups, and these QI-groups, if joined together, will not expose unsafe correlations between identifying and sensitive values. We define two threat levels with regard to the adversary's knowledge about the anonymized data stream.

- **Threat level T_1:** assumes the adversary has no prior knowledge concerning the individuals and the correlations of their identifying and sensitive values in the dataset. She/He is able, however, to extract foreground knowledge from the anonymized dataset that can be used to breach privacy. *For example extracting/knowing renting patterns of individuals, which might lead to link their identifying values to their true identity and track them in the anonymized dataset.*

- **Threat level T_2:** assumes the adversary is equipped with a certain knowledge about the individuals and the correlations of their identifying and sensitive values in the dataset before having access to its anonymized version. She/he can exploit that background knowledge to provoke a privacy breach. *In our renting example, knowing the true identity in plain text of an individual (e.g. Full Name) alongside her/his location patterns might lead to link her/his identity to her/his identifying value in the stream thus exposing her/him in the anonymized stream.*

6.3.1/ PRIVACY MODEL

Given a stream S and two user-defined constants $l \geq 2$ and $k \geq 2$, we say that an anonymization technique safely anonymizes S if it produces a stream S^* that satisfies the following properties:

Property 1 (Safe release of QI-groups)**.** *provides safe correlation of identifying and sensitive values across the released QI-groups such that the intersection of any QI-groups in S^* on their identifying attribute A^{id} yields either k identifying values or none. Formally,*

$\forall v_{id} \in \mathcal{D}(A^{id})$, if $v_{id} \in \pi_{A^{id}} QI_1 \cap ... \cap \pi_{A^{id}} QI_j$, then there exists a set of identifying values $V_{id} \subseteq \mathcal{D}(A^{id})$, such that $V_{id} = \{v_{id}, v_{id_1}, ..., v_{id_{k-1}}\}$ and $V_{id} = \pi_{A^{id}} QI_1 \cap ... \cap \pi_{A^{id}} QI_j$. In other words,

$$\pi_{A^{id}} QI_1 \cap ... \cap \pi_{A^{id}} QI_j = \begin{cases} V_{id} & \text{if } \exists v_{id} \in \pi_{A^{id}} QI_1 \\ & \cap ... \cap \pi_{A^{id}} QI_j \\ \emptyset & \text{otherwise} \end{cases} \tag{6.1}$$

In a less formal definition, the identifying values that are grouped together in a QI-group must always remain grouped together throughout the entire anonymized stream.

Property 2 (l-diverse QI-groups)**.** *ensures that all the anonymized and released QI-groups are l-diverse. Formally,*
$\forall v_{id} \in \mathcal{D}(A^{id}), \forall QI \in S^*, Pr(v_{id}, v_s|QI) \leq 1/l$.

Property 3 (Safe correlation of identifying values)**.** *prohibits linking correlated identifying values in the same QI-group to their corresponding sensitive values, which result in an inherent violation of l-diversity [al Bouna et al., 2013, al Bouna et al., 2015a, Gong et al., 2017]. Formally,*

$\forall v_{id_1}, v_{id_2}, f(v_{id_1}, QI_j) = f(v_{id_2}, QI_j)$ where $f(v_{id_i}, QI_j)$ is a function that returns the number of occurrences of v_{id_i} in QI_j.

Property 3 hides frequent correlations of identifying values in the same QI-groups. It handles cases arising when an adversary may be able to link an individual to his/her sensitive value or to narrow the possibilities for other individuals.

6.3.2/ (k, l)-CLUSTERING FOR PRIVACY PRESERVATION

To preserve our privacy properties, we propose a (k, l)-clustering technique that groups tuples into clusters of disjoint centroids and releases, from these clusters, l-diverse QI-groups containing k distinct identifying values. In brief, our clustering technique works as follows:

- It creates centroids containing k distinct identifying values: $\forall QI_i, QI_j$ two QI-groups released from C, $\pi_{A^{id}} QI_i = \pi_{A^{id}} QI_j = V_{id}(C)$ where $|V_{id}(C)| = k$.

- It ensures that an identifying value exists in one and only centroid: $\forall C_1, C_2\ V_{id}(C_1) \cap V_{id}(C_2) = \emptyset$.

- It releases a QI-group from a cluster C such that: $\forall QI$, a QI-group created from a subset of tuples in the cluster C, and $\forall t \in QI$, $t[A^{id}] \in V_{id}(C)$.

(k, l)-clustering is a bucketization technique that releases l-diverse QI-groups created from a subset of clusters having disjoint centroids. It ensures safe correlation of identifying and sensitive values across the QI-groups, i.e., once k identifying values are grouped in a QI-group, they will remain grouped together in future releases of QI-groups throughout the anonymized stream. We assume that the clustering can be done in two ways, *unsupervised* and *supervised* as defined below.

- **Unsupervised (k, l)-clustering:** has no prior knowledge about the distribution of identifying values in the original dataset. The clustering is done on first-come, first-serve basis inspired by "bottom-up" agglomerative clustering algorithms [Amiri et al., 2016a]. Unsupervised (k, l)-clustering creates cluster centroids and groups tuples accordingly, in reference to their identifying values and privacy constants k and l.

- **Supervised (k, l)-clustering:** has a partial or full view over the distribution of identifying values in the original dataset, thus and unlike the unsupervised clustering, clusters are created based on a predefined set of centroids $\mathcal{V} = \{V_{id}^1, ..., V_{id}^m\}$ that are fed to the clustering technique prior the anonymization. Hence, the identifying and sensitive values that are highly correlated are grouped together in the same cluster to reduce the chances of having these values anonymized/suppressed in order to meet the privacy properties.

As shown in Figure 6.3(c), 'Allen_U1' and 'Cathy_U3' are grouped together in 3 QI-groups because they occur the most in the incoming stream. However in Figure 6.3(b), 'Allen_U1' is grouped alongside 'Betty_U2' and 'Cathy_U3' alongside 'David_U4' due to the order of their tuples in the data stream.

User ID	VIN Number	TS	Location
Allen_U1	abfb32fd10ad2*	1	10.31;17,32
Betty_U2	00983503e35d*	2	30,29;24,12
Cathy_U3	0f550377353d*	3	23,45;11,23
Allen_U1	abfb32fd10ad2*	4	10.32;15,32
David_u4	936e2c77b9du*	5	22,25;11,33
Allen_U1	abfb32fd10ad2*	6	13.32;15,32
Betty_U2	00983503e35d*	7	42,45;11,23
Cathy_U3	0f550377353d*	8	23,45;11,24
Carol_U5	1qwfq2fd10ad2*	9	40,42;14,31
Cathy_U3	0f550377353d*	10	30,30;24,12
...	...	...	...

(a) Incoming Stream S

User ID	VIN Number	GID
Allen_U1	abfb32fd10ad2*	1
Betty_U2	00983503e35d*	1
Cathy_U3	0f550377353d*	2
David_u4	936e2c77b9du*	2
Allen_U1	abfb32fd10ad2*	3
Betty_U2	00983503e35d*	3
...	...	...

GID	Location	TS
1	10.31;17,32	1
1	30,29;24,12	2
2	23,45;11,23	3
2	22,25;11,33	5
3	10.32;15,32	4
3	42,45;11,23	7
...	...	...

(b) Anonymization using Unsupervised *(k,l)*-Clustering

User ID	VIN Number	GID
Allen_U1	abfb32fd10ad2*	1
Cathy_U3	0f550377353d*	1
Allen_U1	abfb32fd10ad2*	2
Cathy_U3	0f550377353d*	2
Allen_U1	abfb32fd10ad2*	3
Cathy_U3	0f550377353d*	3
Betty_U2	00983503e35d*	4
David_U4	936e2c77b9du*	4
...	...	...

GID	Location	TS
1	10.31;17,32	1
1	23,45;11,23	3
2	10.32;15,32	4
2	23,45;11,24	8
3	13.32;15,32	6
3	30,30;24,12	10
4	30,29;24,12	2
4	22,25;11,33	5
...	...	...

(c) Anonymization using Supervised *(k,l)*-Clustering with *V={{U1,U3},{U2,U4},{U5}}*

Figure 6.3: Applying unsupervised and supervised (k, l)-clustering on a data stream with *k,l*=(2,2)

Lemma [1]. *Given a transactional stream S, safe clustering ensures the safe release of QI-groups in the published version S*.*

Proof. Since (k, l)-clustering is applied, $\forall QI_i, QI_j$ two QI-groups released from C, $\pi_{A^{id}} QI_i = \pi_{A^{id}} QI_j = V_{id}(C)$ where $|V_{id}(C)| = k$. Alternatively, since (k, l)-clustering ensures that an identifying value exists in one and only centroid, $\forall C_1, C_2$, two distinct clusters over S*, $V_{id}(C_1) \cap V_{id}(C_2) = \emptyset$ can be written as $\pi_{A^{id}} QI_1 \cap \pi_{A^{id}} QI_2 = \emptyset$ where, QI_1, QI_2 are two QI-groups released respectively from C_1 and C_2. Hence, the intersection of any QI-groups in S* on the identifying values yields either k identifying values or none.

□

6.3.3/ (k, l)-CLUSTERING ALGORITHM

In this section, we present our (k, l)-clustering algorithm applied on a transactional data stream. The main idea behind it is to process incoming tuples on the fly while guarantying safe release of l-diverse QI-groups. It requires two privacy constants k and l, the stream S, and a set of centroids $\mathcal{V}$. (k, l)-clustering outputs an anonymized data stream. The algorithm is composed of two main steps; ”safe clustering” and ”tuple assignment”.

Algorithm 1 (k,l)-clustering(S ,k ,l, $\mathcal{V}$)

Input: $k, l, S, \mathcal{V}$

Output: S^*

/** $\mathcal{V}$ can either be empty or can contain a set of predefined centroids*/

1: C:={};
2: C:= *create_clusters*($\mathcal{V}$); /**Creates $|\mathcal{V}|$ empty clusters and assigns each one a centroid in $\mathcal{V}$ */
3: **while** S is not empty **do**
4: Let t_p be the tuple arrived from S;
5: C_{sel}:=$safe\_clustering(t_p$,$\mathcal{V})$;
6: $tuple\_assignment(t_p$,$C_{sel})$;
7: **end while**

6.3.4/ SAFE CLUSTERING

The function assigns tuples to their corresponding clusters based on their identifying values.

$$t_p \text{ is assigned to } \begin{cases} C_e & \text{if } \exists V_{id}(C_e) \subset \mathcal{V} \text{ where} \\ & t_p[A_{id}] \in V_{id}(C_e) \\ C_q \text{ where } |V_{id}(C_q)| < k & \text{otherwise} \end{cases}$$

1: **function** SAFE_CLUSTERING(t_p,$\mathcal{V}$)
2: *selected_cluster*:={};
3: **if** ($t_p[A_{id}] \notin \mathcal{V}$) **then**
4: C_q:= Find C_q in C where $|V_{id}(C_q)| < k$;
5: **if** $C_q = null$ **then**
6: $V_{id}(C_q)$:={};
7: $V_{id}(C_q) \leftarrow t_p[A_{id}]$; /**Adds $t_p[A_{id}]$ to the empty centroid $V_{id}(C_q)$*/;
8: *selected_cluster*:= C_q;
9: **else**
10: $V_{id}(C_q) \leftarrow t_p[A_{id}]$; /**Adds $t_p[A_{id}]$ to the non-empty centroid $V_{id}(C_q)$ */;
11: *selected_cluster*:= C_q;
12: **end if**
13: **else**
14: Find C_e in C where $t_p[A_{id}] \in V_{id}(C_e)$;
15: *selected_cluster*:= C_e;
16: **end if**
return *selected_cluster*;
17: **end function**

Safe clustering first verifies if the identifying value of the incoming tuple $t_p[A_{id}]$ has been assigned to a centroid, i.e., if it exists in one of the centroids in $\mathcal{V}$. If that is not the case, the algorithm searches for a cluster C_q having a centroid with less than k identifying values

(Steps 3-4). If C_q exists, the new identifying value $t_p[A_{id}]$ is added to the centroid $V_{id}(C_q)$ and the cluster C_q is returned (Steps 10-11). However, if C_q is empty, a new centroid is created for $t_p[A_{id}]$ and the empty cluster C_q is returned (Steps 5 to 8). Now, if $t_p[A_{id}]$ has been already assigned to a centroid, safe clustering returns its corresponding cluster C_e (Steps 14-15).

6.3.5/ TUPLE ASSIGNMENT

It assigns a tuple t_p to the selected cluster C_{sel} as follows: In a given cluster, all tuples are distributed over multiple *sub-groups*. The same identifying value does not appear twice in the same *sub-group* (steps 3-4-5). A *sub-group* must contain k tuples before verifying its l-diversity (step 6-7). Each *sub-group* is published as two separate tables QI_{table} and SV_{table} linked by the same *GroupID* (GID) in case the l-diversity property is verified (steps 8-9). Otherwise, the *sub-group* is added to *temp* which is a unique structure for each cluster that combines all the non l-diverse *sub-groups* (Steps 10-11). Each time we add a non l-diverse *sub-group* to *temp*, its l-diversity is tested (Steps 12-13). In addition, the procedure empties *temp* after publishing it.

```
1: procedure TUPLE_ASSIGNMENT(t_p, C_sel)
2:     sub-group:={};
3:     sub-group:=Find largest sub-group in C_sel.subgroups[] where t_p[A_id] ∉
   π_{A^id} sub-group;
4:     if sub-group ≠ null then
5:         sub-group ← t_p; /**Add t_p to sub-group*/
6:         if (sub-group.size = k) then
7:             if (sub-group is l-diverse) then
8:                 Publish sub-group as QI_table and SV_table linked by GID
9:                 Delete sub-group
10:            else
11:                temp := temp ⋃ sub-group
12:                if (temp.size > k and temp is l-diverse) then
13:                    Publish temp as QI_table and SV_table linked by GID
14:                    Delete temp
15:                end if
16:            end if
17:        end if
18:    else
19:        sub-group ← t_p;
20:        subgroups[] ← sub-group; /**Add sub-group to the rest of the non-published
   subgroups in the cluster*/
21:    end if
22: end procedure
```

After processing the entire stream, the algorithm will publish all *temporary sub-groups* (stored in the *temp* structure), i.e. the groups which are not l-diverse nor reached size k, by suppressing the identifying values. This guarantees the privacy constraint but impacts the utility of the dataset. This temporary *sub-group* in each cluster will have the same GID key once published, no matter its size.

6.4/ EXPERIMENTS

In this section, we evaluate the efficiency of our unsupervised and supervised (k, l)-clustering techniques by conducting a set of experiments detailed hereinafter. The algorithm is implemented in JAVA and tested on a PC with 2.20 GHz Intel Core i7 CPU, 8.0 GB RAM.

Input Data Stream & Relational Schema To simulate a data stream scenario, we used a rental transaction dataset composed of 109763 tuples where each tuple is associated with a timestamp. We assume that at each time instant exactly one tuple arrives. As a result, timestamps range from 1 to |S|. The dataset contains 2374 distinct identifying values.

We designed two different sets of experiments in order to examine the effectiveness of our approach in terms of utility by:

- Evaluating the percentage of suppressed identifying values.
- Evaluating the delay-retention of tuples in the queue before being released in QI-groups.

As previously stated, after processing the stream over a specified interval of time, our algorithm suppresses the identifying values in the QI-groups that are not l-diverse nor of size k.

Percentage of suppressed identifying values Using the unsupervised (k, l)-clustering, we vary the value of k from 3 to 8, and examine the percentage of suppressed values. The parameter l is set to 3. For high values of k, the percentage of suppressed values increases. It reaches almost 60% for k=8 as shown in Figure 6.4. Here, we cluster identifying values based on their order of arrival. Each k individuals clustered together might not have the same distribution over the stream. Therefore when k increases, it becomes more difficult to form QI-groups leading to an increase in the amount of suppressed values.

Using the supervised (k, l)-clustering we ensure that the most frequent identifying values are clustered then grouped together in the QI-groups. Consequently, we suppress fewer identifying values and thus, obtain better utility, as shown in Figure 6.4, where the percentage of suppressed values reaches 1% for k=20.

Retention of tuples A tuple is retained in the queue if it remains a) in a *sub-group* that did not reach size k or b) in the *temporary sub-group* of the corresponding cluster.

To determine the average retention delay, we run our algorithm multiple times while varying k between 3 and 8 for both approaches. For each set of $\{k,l\}$ values, we measure the retention delay of each tuple in memory. Then we compute the average time of all the tuples. For both methods, the average value falls in the range of 1 to 2 seconds. This value is chosen as the delay constraint δ defined in [Pervaiz et al., 2015]

We consider a tuple that remains more than the specified delay δ in the memory a "delayed or outdated tuple". δ slightly varies with k. We applied our algorithm on the same

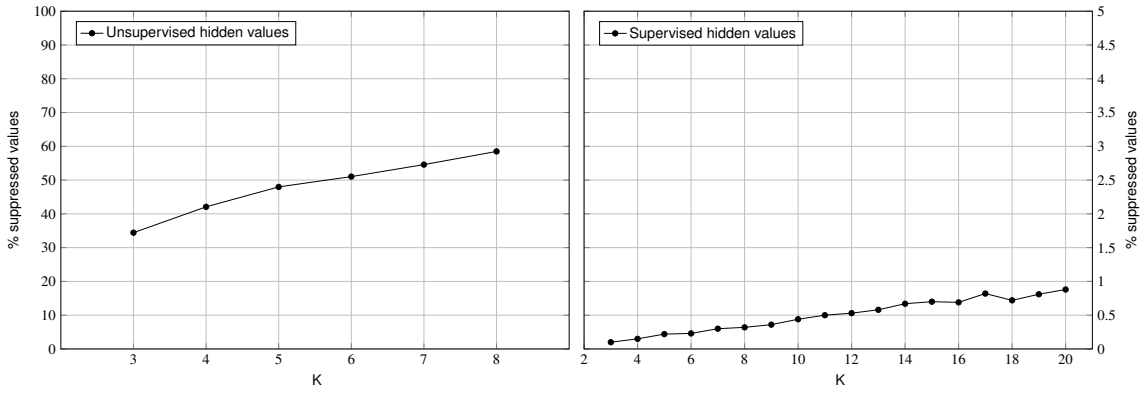


Figure 6.4: Percentage of suppressed values for $l = 3$ while varying k for both Unsupervised and Supervised (k, l)-clustering approaches

rental dataset we used before, while adopting both approaches, as shown in Figure 6.5. The delay constraint can be chosen according to the data stream application requirement regarding availability of the anonymized tuples as stated in [Pervaiz et al., 2015].

6.5/ RELATED WORKS

In [Cao et al., 2011], Cao et al. extend the definition of k-anonymity to apply it on data streams and propose CASTLE, a clustering-based algorithm that publishes k-anonymized clusters in an acceptable delay. An extension of CASTLE is presented in [Pervaiz et al., 2015] to reduce the number of tuples in the clusters and to maximize the utility of the anonymized dataset. In another work [Zakerzadeh et al., 2011], FAANST is proposed to anonymize numerical data streams. It achieves suitable processing time with good trade-off between utility and privacy. However, while applying FAANST, some tuples might be stuck in memory and expire. Hence, the authors in [Zakerzadeh et al., 2013] proposed two approaches that define a soft deadline for each tuple in memory, and if a tuple stays more than the specified deadline in the system, the algorithms force the tuple to be published. FADS is an anonymization algorithm proposed in [Mohammadian et al., 2014, Guo et al., 2013] that has convenient time and space scale with additional constraints on the size of the clusters size and their reuse strategy. Also in another study [Mohamed et al., 2016], the authors proposed a clustering approach for k-anonymizing distributed data streams generated by different sites. The authors performed the clustering locally on each site and then shared the local clusters with a global server in order to construct the global cluster and publish the anonymized stream. Also, the authors in [Sopaoglu et al., 2020] proposed a tunable algorithm UBDSA in which the importance of loss in terms of data latency and quality can be adjusted. The authors in [Wang et al., 2018] extend ρ–uncertainty [Cao et al., 2010] and apply it to a transactional data stream. On another note, the authors in [Wang et al., 2010]

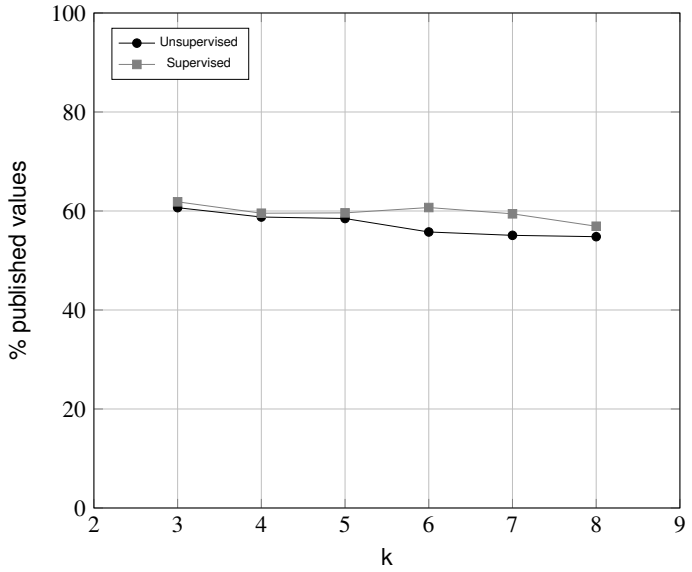


Figure 6.5: Percentage of published tuples for both approaches before δ

applied the bucketization-based technique, anatomy [Xiao et al., 2006a], in the context of data stream. While these techniques extend privacy solutions mainly based on k-anonmyity [Sweeney, 2002] and l-diversity[Machanavajjhala et al., 2006] in the context of data streams, they do not take into account the correlation of the identifying and sensitive values across the QI-groups due to the transactional nature of the data. In this work, we assume that identifying attributes along with non-sensitive and sensitive attributes must be combined together and stored in the dataset for analysis purposes. Moreover, several studies [Kifer, 2009b, Wong et al., 2011b, Gong et al., 2017, Amiri et al., 2018] have shown that correlations attacks can be launched not only on bucketization-based techniques but on generalization-based techniques as well. In [Domingo-Ferrer et al., 2019], the authors propose a generalization-based micro-aggregation algorithm for stream k-anonymity that meets a maximum delay constraint, without preserving the order of incoming tuples in the published stream such as in [Cao et al., 2011]. Then, they improve the preservation of the original order of the tuples by using steered micro-aggregation while adding the timestamp as an artificial attribute. Similar to [Cao et al., 2011], we do not publish the time stamp attribute due to privacy constraints however we use it for experimental purposes.

A similar work to ours is defined in [Amiri et al., 2016b] where the authors include background knowledge in their anonymization algorithm to deal with strong adversaries able to associate the sensitive values to their owners based on quasi-identifying values. They propose a hierarchical agglomerative algorithm that works in two separate phases. The first generates clusters of tuples and the second enforces the privacy constraints, namely k-anonymity, β-likeness, and l-diversity, in order to prevent attribute and identity disclosure. By doing so, the authors only address one facet of the problem, which is the one

related to correlations known to the adversary. Here, we consider that the correlations can be mined from the dataset and used as foreground knowledge to link individuals to their sensitive values. Also the authors in [Amiri et al., 2016b] worked in the context of static tabular datasets whereas in our case we are acting against the correlation in the context of a data stream. On the one hand, the authors in [Amiri et al., 2018] propose a knowledge-based sequential anonymization algorithm (KSAA) for privacy preservation in the context of a sequential data publishing scenario[1]. They present a bottom-up anonymization algorithm, KSAA that uses generalization to protect against background knowledge attacks. KSAA clusters tuples and generates QI-groups satisfying the privacy model in the current view. It checks, in a second step, if the privacy constraint is satisfied when several views are joined together. On the other hand, the authors in [Amiri et al., 2019, Riboni et al., 2012] address the correlation problem by proposing generalization-based techniques in the context of continuous data publishing scenario[2]. The proposed algorithms in [Amiri et al., 2018, Amiri et al., 2019, Riboni et al., 2012] tackle the correlation problem however not in a data stream scenario where three requirements must be met including low retention of tuples, balanced memory usage and run time. On another note, several noticeable works [Bonomi et al., 2016, Zhang et al., 2017, Nie et al., 2016] have been done for differential privacy [Dwork et al., 2016] for streaming data. In this work, we choose to work with bucketization-based technique that releases trustworthy information. We particularly extend previous works [al Bouna et al., 2013, al Bouna et al., 2015a] to address correlations in the data stream in data sharing scenarios.

6.6/ CONCLUSION

In this chapter, we defined new privacy properties in order to address the correlation problem in the anonymization of a transactional data stream. A bucketization-based technique, entitled (k, l)-clustering, is proposed to enforce these privacy properties. (k, l)-clustering processes incoming tuples on the fly. It continuously groups k distinct individuals into l-diverse QI-groups and ensures that these individuals remain grouped together in future releases of QI-groups. We evaluated our algorithm in terms of utility and time complexity by considering two approaches: *supervised* and *unsupervised*. We showed, by conducting a set of experiments, that both approaches cope well with the streaming nature of the data while respecting the privacy constraints. The *supervised* approach yielded better results due to the fact that it has a partial or full view over the distribution of identifying values in the dataset.

[1] A sequential publishing scenario is when we publish different views/versions of a table where each one may contain a different subset of attributes, i.e. the set of published attributes change over the different views.

[2] A continous publishing scenario is when we continuously release new tuples that belong to the dataset while preserving the same schema (i.e., the same attributes).

IV

General Conclusion

7

General Conclusion

7.1/ Closing words

7.1.1/ Abundance of data

Almost every product that we use throughout our daily lives generates and releases a certain type of data (e.g., digital images, relational data, sensory data...). As mentioned in [Schneier, 2015], "we as a species were generating by 2010 more data per day than we did from the beginning of time until 2003". Furthermore, the generated data depicts us, the product users/the data subjects, in one way or another. It can monitor and record our behavior when it comes to breathing, eating, driving, talking/chatting with our family and friends, doing sports, sleeping ... On the one hand, this huge amount of information (i.e. Big Data) that is gathered and stored can indeed enhance the quality of the products that we use and make our lives easier: *for instance, recommending the next book to read or which restaurant to go to next or which turn to take while driving*. On the other hand, if we look at it from a different perspective, the gathering, storing and processing of this information is complementary to the definition of the surveillance concept: "The act of observing persons or groups either with notice or their knowledge (overt surveillance) or without their knowledge (covert surveillance)"[1].

7.1.2/ Privacy and anonymity

"In the privacy of our home or bedroom, we can relax in a way that we can't when someone else is around. Privacy is an inherent human right, and a requirement for maintaining the human condition with dignity and respect. It is about choice and having the power to control how you present yourself to the world."
~ Bruce Schneier (Data and Goliath [Schneier, 2015], p. 148).

[1] https://www.law.cornell.edu/wex/surveillance

First of all, privacy is an essential right for every human being [Committee, 1948]. Processing data without taking into consideration the data subject's privacy is a breach to her/his essential rights, hence a crime. Second of all, committing this crime has repercussions on individuals' liberties and freedom of choice as well as on democratic referendums and elections (Cambridge Analytica scandal [Cadwalladr et al., 2018]). Automated systems (recommendation systems or information filtering systems) that are embedded almost in every platform/product nowadays can affect the user's behavior from purchase choices (by recommending certain products based on the user's location) to political views (by filtering the news the user receives based on her/his profile data, location or social connections). Specifically in the latter case, data misconducts and privacy breaches can manipulate and mislead societies to fit/accomplish certain goals that are sometimes not in their best interest. History showed us multiple times the dangerous repercussions of mis-lead and indoctrinated societies.

As a result, several data protection regulations were introduced in the last couple of years such as the GDPR [European Parliament, 2018b] in Europe and CCPA [Legislature, 2018] in the United States. These regulations compel data-driven organizations and companies to process data in a responsible and compliant way by respecting and preserving the rights of the data subjects including their privacy and anonymity. One way to preserve these rights and protect personal data against privacy breaches is to apply privacy preserving techniques such as data anonymization[2]. As stated in [Schneier, 2015], "anonymity protects privacy, it empowers individuals and it is fundamental to liberty".

7.1.3/ Re-evaluating anonymization techniques

"The very notion of anonymization needs to be revisited, as the adversarial models are evolving and, thus, anonymization is becoming more and more challenging in real case scenarios."
~ European Union Agency for Cybersecurity ENISA (Pseudonymisation techniques and best practices [Jensen et al., 2019], p. 43).

Multiple data anonymization techniques have been proposed, developed, and adopted throughout the years. However, as mentioned in the ENISA technical report [Jensen et al., 2019], anonymization techniques and privacy models should always be

[2]Several other steps should be taken as well from getting consent from the user to limiting the amount of data that is being gathered and processed.

re-evaluated as the adversaries are evolving and becoming more challenging. New attacking scenarios regarding the adversary's goal, knowledge and capabilities should be studied to re-assess anonymization techniques and in some cases enhance them to try and guarantee the anonymity of the data subjects. Therefore, throughout our thesis, we (i) propose and implement several anonymization techniques and tools in the context of relational data streams and images and (ii) assess the robustness of these techniques by simulating adversaries with different knowledge and several attacking capabilities. In addition, we designed our proposed frameworks to be scalable because new attacking capabilities as well as new anonymization techniques should be considered in the future.

7.2/ SUMMARY OF THE DIFFERENT CONTRIBUTIONS

In this section, we summarize our contributions and we discuss briefly the experimental results that we conducted throughout this study.

7.2.1/ THE FIRST CONTRIBUTION

In Chapter 3, we designed and implemented an anonymization tool that localizes and obfuscates identifying/sensitive information in images/videos via DL-based techniques. Several anonymization tools are available today on the market [sightengine, 2020, brighter AI, 2019, eyedea Regonition, 2021, Celantur,] however not a single one combines the following features: (i) scalable in terms of obfuscation techniques, (ii) agnostic in terms of localization approaches, (iii) modular in terms of sensitive information, (iv) GDPR compliant, (v) open-source and (vi) BMW compatible.

7.2.2/ THE SECOND CONTRIBUTION

In Chapter 4, we proposed a generic and scalable framework to evaluate and recommend the most robust obfuscation techniques for specific identifying/sensitive information, such as an individual's face. The framework reconstructs/recognizes obfuscated faces via DL-assisted attacks, evaluates the reconstruction/recognition via different metrics and recommends the most robust obfuscation with regard to each metric. We presented the recommendation framework by (i) proposing a 4-layered iterative process, (ii) showcasing the framework's detailed structure when applied to a facial images dataset and (iii) defining the 3-components adversary model (goal, knowledge and capabilities) to our application domain (i.e., facial features obfuscations) with two threat levels and three attacking capabilities. We conducted three sets of experiments on the CelebA dataset [Liu et al., 2015]. In the first experiment, we validated our approach by implementing and testing our frame-

work on obfuscated faces. Throughout the second experiment, we demonstrated how the adversary's attacking capabilities scale with her/his knowledge and how it increased the potential risk of breaching the identities of blurred face images. In the third experiment, we studied the possible privacy breaches and the attack range of an adversary against blurred face images while lacking knowledge about the obfuscation's hyper-parameters.

7.2.3/ THE THIRD CONTRIBUTION

In Chapter 5, we empirically demonstrated that Federated Learning (FL) can be used as a collaborative attack/adversarial strategy to (i) remedy the lack of background knowledge and data shortage, (ii) leverage the attacking capabilities of an adversary and (iii) increase the privacy breaches without the need to share/disclose the local training datasets in a centralized location. We defined seven collective threat levels based on the background knowledge of the different adversaries and the sharing of that knowledge. We conducted four experimental use cases on the Face Scrub dataset. Throughout the four use cases, we showed that FL leverages the capabilities of each adversary participating in the FL-assisted attack despite data shortage and the lack of background knowledge. For instance, in one threat level, an adversary, with no knowledge about the obfuscation technique, leveraged the recognition-based attack against blurred face images and re-identified 57 out of 100 individuals instead of just 1 while only sharing the model's parameters. Second, we demonstrated that increasing the number of adversaries participating in the FL-assisted attack leads to more serious attacks and intensifies the privacy breaches. For instance, when 2 adversaries collaborated together, the average number of recognized individuals was 50 out of 100 (50.8%) whereas when 5 adversaries collaborated together, the average number was 88 out of 100 (71.2%). Last but not least, we showed that sharing the background knowledge between the adversaries increases significantly the attacking capabilities. For instance, the adversaries were able to recognize the identities of 58 individuals (53%) on average without sharing their knowledge, whereas they recognized 88 individuals (72%) when sharing it.

7.2.4/ THE FOURTH CONTRIBUTION

In Chapter 6, we have defined new privacy properties in order to address the correlation problem in the anonymization of a transactional data stream. We proposed a bucketization-based technique, entitled (k, l)-clustering, to enforce the privacy properties. (k, l)-clustering processes incoming tuples on the fly. It continuously groups k distinct individuals into l-diverse QI-groups and ensures that these individuals remain grouped together in future releases of QI-groups. We evaluated our algorithm in terms of utility and time complexity by considering two approaches: *supervised* and *unsupervised*. We

showed, by conducting a set of experiments, that both approaches cope well with the streaming nature of the data while respecting the privacy constraints. The *supervised* approach yielded better results due to the fact that it has a partial or full view over the distribution of identifying values in the dataset.

7.3/ LIMITATIONS AND FUTURE WORKS

In this section, we note some of the limitations of our work and we try to propose possible approaches to address them in future works.

7.3.1/ IMPROVING THE FIRST CONTRIBUTION

In Chapter 3, we considered the three traditional obfuscation techniques e.g., pixelating, masking, and blurring. However, other obfuscation techniques such as the GAN-based inpainting approaches [Hao et al., 2019] or k-same methods [Newton et al., 2005a] might provide higher privacy levels while preserving relevant features in the image. Therefore, adding and adapting such obfuscation techniques should be considered in future works.

7.3.2/ IMPROVING THE SECOND CONTRIBUTION

Prospects that we did not explore in Chapter 4 could be addressed in future work:

- Other visual features such as an individual's name tag, posture or personal belongings can be identifying and considered sensitive. In this work, we focused on individuals' faces because they are the most revealing in the context of images.
- The adversary's background knowledge covered the identities $present$ in the target dataset, therefore she/he can mine images for each known identity and perform a DL-assisted attack to recognize and re-identify the identity of the obfuscated face images. Nevertheless, in other scenarios the adversary's background knowledge could be limited to quasi-identifying information such as the individual's race or gender. If that is the case, the adversary could perform DL-assisted attacks to recognize the gender or the race of the target individual instead of the full identity which might lead as well to potential privacy breaches when linked to other data sources (i.e., identity disclosure via linking attacks).
- Different approaches have been proposed in the context of image classification and identity recognition to trick, ruin or corrupt DL models. Some approaches rely on designing adversarial examples by perturbing the query image at the inference phase

either physically (e.g. the target individual wears special accessories, e.g. glasses or hats [Komkov et al., 2021]) or quantitatively [Goodfellow et al., 2014] (small perturbations are added on a pixel level which are not visible to the human visual system). Other approaches rely on modifying/corrupting the training dataset via data poisoning (clean-label [Shafahi et al., 2018, Shan et al., 2020] and dirty-label attacks [Biggio et al., 2012]), to ruin the neural network's weights and trick it into inferring incorrect labels when queried with non-perturbed images, i.e., breaking the DL models at training phase. These approaches can be employed in our scenario as a defense mechanism against the adversary's attempts to breach the obfuscated faces' anonymity. Nevertheless, it requires a thorough examination and investigation therefore we leave the defender concept for a future study.

7.3.3/ IMPROVING THE THIRD CONTRIBUTION

There are also some limitations in Chapter 5 that could be addressed in future work.

- We did not consider other obfuscations such as pixelating or masking a person's face in the target dataset. Furthermore, we can consider studying other types of DL-assisted attacks such as *restoration*-based attacks.

- It is vital to extensively study the defender concept and how to counteract against the FL-assisted attack. Different approaches can be employed such as data poisoning [Biggio et al., 2012] or model poisoning [Bagdasaryan et al., 2020] to try and decrease the accuracy of the FL-assisted attack. Also as mentioned in Section 5.2, in this work we adapted the *standard* federated averaging algorithm which does not perform well when the local datasets are heterogeneous. Therefore as mentioned in Section 5.5.3, additional approaches [Zhao et al., 2018, Hsu et al., 2020, Hsu et al., 2019, Hsieh et al., 2020] could be adapted/applied to remedy these limitations, hence perform a stronger FL-assisted attack.

7.3.4/ IMPROVING THE FOURTH CONTRIBUTION

In Chapter 6, we proposed the (k, l)-clustering algorithm to address the correlation problem while anonymizing a transactional data stream. Each tuple within the stream had three attributes: VIN (i.e., ID), location and time stamp. Considering another dataset with additional attributes (e.g., quasi-identifying attributes) increases the attack surface of the adversaries. Hence, we need to adapt/improve our algorithm to respect and preserve the privacy constraints (e.g. k-anonymity) when considering additional attributes.

7.4/ CONCLUSION

In this thesis, we (i) proposed and implemented several anonymization techniques and tools in the context of images and relational data streams and (ii) assessed the robustness of these techniques by simulating adversaries with different knowledge and several attacking capabilities.

Designing, developing and implementing new privacy preserving techniques (e.g. obfuscation techniques for images or perturbative/non-perturbative techniques for relational datasets) is important. Similarly, simulating adversaries with different background knowledge to re-evaluate and assess the validity and robustness of these techniques is equally vital. These two concepts, i.e. defender and adversary, should always be investigated simultaneously. For instance regarding the obfuscation techniques in the context of images, we believe that in the coming years, the GAN-based techniques [Hao et al., 2019] will/should gain more momentum. These techniques modify the images' features while maintaining the semantic information: *For instance, these techniques can obfuscate an individual's face by replacing the original facial features (e.g., eyes, mouth, lips, nose) with synthetic ones...* The GAN-based techniques [Hao et al., 2019] techniques can/should preserve the anonymity of the data subjects while maintaining better utility in comparison with the current mainstream obfuscation techniques (e.g. blurring, pixelating, or masking) [Caesar et al., 2020, Frome et al., 2009]. However, further quantitative and qualitative evaluations similar to the ones performed in Chapter 4 and [Li et al., 2017b, Dufaux et al., 2010, Nawaz et al., 2017] should be conducted to measure the validity and robustness of these techniques.

On another note, the different scientific contributions constituting this report are the following:

- **Image Obfuscation tool at BMW Group**: this chapter was published as a public GitHub Repository [Tekli et al., 2021], as a press release by BMW Group [Hatzel, 2021] and as a white paper by Intel Cooperation [Intel, 2021]
- **A Framework for Evaluating Image Obfuscation under Deep Learning-Assisted Privacy Attacks**: this chapter was published partially in the 17th International Conference on Privacy, Security and Trust (PST) in 2019 [Tekli et al., 2019] and is currently under peer review in a scientific journal [Tekli et al., NDb].
- **Leveraging Deep Learning-Assisted Attacks against Image Obfuscation via Federated Learning**: this chapter is currently under peer review in a scientific journal [Tekli et al., NDa].
- ***(k,l)*-Clustering for Transactional Data Streams Anonymization**: this chapter

was published in the International Conference on Information Security Practice and Experience ISPEC 2018 [Tekli et al., 2018].

BIBLIOGRAPHY

[Abramian et al., 2019] Abramian, D., et Eklund, A. (2019). **Refacing: Reconstructing anonymized facial features using GANS**. In *16th IEEE International Symposium on Biomedical Imaging, ISBI 2019, Venice, Italy, April 8-11, 2019*, pages 1104–1108. IEEE.

[Agarap, 2018] Agarap, A. F. (2018). **Deep learning using rectified linear units (relu)**. *arXiv preprint arXiv:1803.08375*.

[Ahonen et al., 2006] Ahonen, T., Hadid, A., et Pietikainen, M. (2006). **Face description with local binary patterns: Application to face recognition**. *IEEE transactions on pattern analysis and machine intelligence*, 28(12):2037–2041.

[AI, 2018] AI, U. (2018). **understand.ai anonymizer**. https://github.com/understand-ai/anonymizer.

[al Bouna et al., 2013] al Bouna, B., Clifton, C., et Malluhi, Q. M. (2013). **Using safety constraint for transactional dataset anonymization**. In *DBSec*, pages 164–178.

[al Bouna et al., 2015a] al Bouna, B., Clifton, C., et Malluhi, Q. M. (2015a). **Anonymizing transactional datasets**. *Journal of Computer Security*, 23(1):89–106.

[al Bouna et al., 2015b] al Bouna, B., Clifton, C., et Malluhi, Q. M. (2015b). **Efficient sanitization of unsafe data correlations**. In *Proceedings of the Workshops of the EDBT/ICDT 2015 Joint Conference (EDBT/ICDT), Brussels, Belgium, March 27th, 2015.*, pages 278–285.

[Amidi, 2019] Amidi, S. (2019). **Cs 230 - deep learning**.

[Amiri et al., 2018] Amiri, F., Yazdani, N., et Shakery, A. (2018). **Bottom-up sequential anonymization in the presence of adversary knowledge**. *Information Sciences*, 450:316–335.

[Amiri et al., 2016a] Amiri, F., Yazdani, N., Shakery, A., et Chinaei, A. H. (2016a). **Hierarchical anonymization algorithms against background knowledge attack in data releasing**. *Know.-Based Syst.*, 101(C):71–89.

[Amiri et al., 2016b] Amiri, F., Yazdani, N., Shakery, A., et Chinaei, A. H. (2016b). **Hierarchical anonymization algorithms against background knowledge attack in data releasing**. *Knowl. Based Syst.*, 101:71–89.

[Amiri et al., 2019] Amiri, F., Yazdani, N., Shakery, A., et Ho, S. (2019). **Bayesian-based anonymization framework against background knowledge attack in continuous data publishing**. *Trans. Data Priv.*, 12(3):197–225.

[Amos et al., 2016] Amos, B., Ludwiczuk, B., et Satyanarayanan, M. (2016). **Openface: A general-purpose face recognition library with mobile applications**.

[Anjum et al., 2017] Anjum, A., et Raschia, G. (2017). **Banga: an efficient and flexible generalization-based algorithm for privacy preserving data publication**. *Computers*, 6(1):1.

[Arivazhagan et al., 2019] Arivazhagan, M. G., Aggarwal, V., Singh, A. K., et Choudhary, S. (2019). **Federated learning with personalization layers**. *CoRR*, abs/1912.00818.

[Arulkumaran et al., 2017] Arulkumaran, K., Deisenroth, M. P., Brundage, M., et Bharath, A. A. (2017). **Deep reinforcement learning: A brief survey**. *IEEE Signal Processing Magazine*, 34(6):26–38.

[Awad, 2020] Awad, N. (2020). **Publishing set-valued dataset strengthening the dis-association approach to improve both privacy preservation and utility**.

[Ayle et al., 2020] Ayle, M., Tekli, J., Zini, J. E., Asmar, B. E., et Awad, M. (2020). **Bar - a reinforcement learning agent for bounding-box automated refinement**. In *AAAI*.

[Bagdasaryan et al., 2020] Bagdasaryan, E., Veit, A., Hua, Y., Estrin, D., et Shmatikov, V. (2020). **How to backdoor federated learning**. In Chiappa, S., et Calandra, R., editors, *The 23rd International Conference on Artificial Intelligence and Statistics, AISTATS 2020, 26-28 August 2020, Online [Palermo, Sicily, Italy]*, volume 108 of *Proceedings of Machine Learning Research*, pages 2938–2948. PMLR.

[Barbaro et al., 2006] Barbaro, M., et Zeller, T. (2006). **A face is exposed for aol searcher no. 4417749**.

[Bay et al., 2006] Bay, H., Tuytelaars, T., et Van Gool, L. (2006). **Surf: Speeded up robust features**. In *European conference on computer vision*, pages 404–417. Springer.

[Bayardo et al., 2005] Bayardo, R. J., et Agrawal, R. (2005). **Data privacy through optimal k-anonymization**. In *21st International conference on data engineering (ICDE'05)*, pages 217–228. IEEE.

[Belhumeur et al., 1997] Belhumeur, P. N., Hespanha, J. P., et Kriegman, D. J. (1997). **Eigenfaces vs. fisherfaces: Recognition using class specific linear projection**. *IEEE Transactions on pattern analysis and machine intelligence*, 19(7):711–720.

[Bellare et al., 2000] Bellare, M., Pointcheval, D., et Rogaway, P. (2000). **Authenticated key exchange secure against dictionary attacks**. *IACR Cryptol. ePrint Arch.*, 2000:14.

[Bellare et al., 1993a] Bellare, M., et Rogaway, P. (1993a). **Entity authentication and key distribution**. In Stinson, D. R., editor, *Advances in Cryptology - CRYPTO '93, 13th Annual International Cryptology Conference, Santa Barbara, California, USA, August 22-26, 1993, Proceedings*, volume 773 of *Lecture Notes in Computer Science*, pages 232–249. Springer.

[Bellare et al., 1993b] Bellare, M., et Rogaway, P. (1993b). **Entity authentication and key distribution**. In Stinson, D. R., editor, *Advances in Cryptology - CRYPTO '93, 13th Annual International Cryptology Conference, Santa Barbara, California, USA, August 22-26, 1993, Proceedings*, volume 773 of *Lecture Notes in Computer Science*, pages 232–249. Springer.

[Bellare et al., 1995a] Bellare, M., et Rogaway, P. (1995a). **Provably secure session key distribution: the three party case**. In Leighton, F. T., et Borodin, A., editors, *Proceedings of the Twenty-Seventh Annual ACM Symposium on Theory of Computing, 29 May-1 June 1995, Las Vegas, Nevada, USA*, pages 57–66. ACM.

[Bellare et al., 1995b] Bellare, M., et Rogaway, P. (1995b). **Provably secure session key distribution: the three party case**. In Leighton, F. T., et Borodin, A., editors, *Proceedings of the Twenty-Seventh Annual ACM Symposium on Theory of Computing, 29 May-1 June 1995, Las Vegas, Nevada, USA*, pages 57–66. ACM.

[Bengio, 2009] Bengio, Y. (2009). **Learning deep architectures for AI**. Now Publishers Inc.

[Biggio et al., 2012] Biggio, B., Nelson, B., et Laskov, P. (2012). **Poisoning attacks against support vector machines**. In *Proceedings of the 29th International Conference on Machine Learning, ICML 2012, Edinburgh, Scotland, UK, June 26 - July 1, 2012*. icml.cc / Omnipress.

[Board, 2019] Board, E. D. P. (2019). **The danish data protection agency proposes a dkk 1,2 million fine for danish taxi company**. https://edpb.europa.eu/news/national-news/2019/danish-data-protection-agency-proposes-dkk-12-million-fine-danish-taxi_en. Accessed: 2021.

[Bonomi et al., 2016] Bonomi, L., et Xiong, L. (2016). **On differentially private longest increasing subsequence computation in data stream**. *Trans. Data Priv.*, 9(1):73–100.

[Boracchi et al., 2012] Boracchi, G., et Foi, A. (2012). **Modeling the performance of image restoration from motion blur**. *IEEE Trans. Image Process.*, 21(8):3502–3517.

[Bouna et al., 2015] Bouna, B. A., Clifton, C., et Malluhi, Q. (2015). **Efficient sanitization of unsafe data correlations**. In *Proc. the Workshops of the EDBT/ICDT Joint Conf*, pages 278–285. Citeseer.

[brighter AI, 2019] brighter AI (2019). **Brighter ai protect every identity on roads**. https://brighter.ai/.

[Cadwalladr et al., 2018] Cadwalladr, C., et Graham-Harrison, E. (2018). **Revealed: 50 million facebook profiles harvested for cambridge analytica in major data breach**. https://www.theguardian.com/news/2018/mar/17/cambridge-analytica-facebook-influence-us-election.

[Caesar et al., 2020] Caesar, H., Bankiti, V., Lang, A. H., Vora, S., Liong, V. E., Xu, Q., Krishnan, A., Pan, Y., Baldan, G., et Beijbom, O. (2020). **nuscenes: A multimodal dataset for autonomous driving**.

[Cambridge, 1994] Cambridge, A. L. (1994). **Laboratories cambridge. the database of faces**. https://cam-orl.co.uk/facedatabase.html.

[Campan et al., 2011] Campan, A., Cooper, N., et Truta, T. M. (2011). **On-the-fly generalization hierarchies for numerical attributes revisited**. In *Workshop on Secure Data Management*, pages 18–32. Springer.

[Cao et al., 2011] Cao, J., Carminati, B., Ferrari, E., et Tan, K. (2011). **Castle: Continuously anonymizing data streams**. *IEEE Transactions on Dependable and Secure Computing*, 8(3):337–352.

[Cao et al., 2010] Cao, J., Karras, P., Raïssi, C., et Tan, K. (2010). **rho-uncertainty: Inference-proof transaction anonymization**. *Proc. VLDB Endow.*, 3(1):1033–1044.

[Celantur,] Celantur. **Privacy-preserving image and video blurring**. https://www.celantur.com/.

[chapman & Hall Book, 2016] chapman & Hall Book, A. (2016). **Data Privacy: Principles and Practice**. Chapman and Hall/CRC.

[Chen et al., 2018] Chen, L., Papandreou, G., Kokkinos, I., Murphy, K., et Yuille, A. L. (2018). **Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs**. *IEEE Trans. Pattern Anal. Mach. Intell.*, 40(4):834–848.

[Chen et al., 2017] Chen, L.-C., Papandreou, G., Kokkinos, I., Murphy, K., et Yuille, A. L. (2017). **Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs**. *IEEE transactions on pattern analysis and machine intelligence*, 40(4):834–848.

[Chen et al., 2008] Chen, Y., et Chu, W. W. (2008). **Protection of database security via collaborative inference detection**. *IEEE Trans. Knowl. Data Eng.*, 20(8):1013–1027.

[Ciriani et al., 2010] Ciriani, V., Vimercati, S. D. C. D., Foresti, S., Jajodia, S., Paraboschi, S., et Samarati, P. (2010). **Combining fragmentation and encryption to protect privacy in data storage**. *ACM Trans. Inf. Syst. Secur.*, 13:22:1–22:33.

[Committee, 1948] Committee, D. (1948). **Universal declaration of human rights**. https://www.un.org/en/about-us/universal-declaration-of-human-rights.

[Cordts et al., 2016] Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., Franke, U., Roth, S., et Schiele, B. (2016). **The cityscapes dataset for semantic urban scene understanding**. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3213–3223.

[Cormode et al., 2010] Cormode, G., Srivastava, D., Li, N., et Li, T. (2010). **Minimizing minimality and maximizing utility: analyzing method-based attacks on anonymized data**. *Proceedings of the VLDB Endowment*, 3(1-2):1045–1056.

[Courtiol et al., 2019] Courtiol, P., Maussion, C., Moarii, M., Pronier, E., Pilcer, S., et et al. (2019). **Deep learning-based classification of mesothelioma improves prediction of patient outcome**. *Nature medicine*, 25(10):1519—1525.

[Dalal et al., 2005] Dalal, N., et Triggs, B. (2005). **Histograms of oriented gradients for human detection**. In *2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05)*, volume 1, pages 886–893. Ieee.

[Deng et al., 2009] Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., et Fei-Fei, L. (2009). **Imagenet: A large-scale hierarchical image database**. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee.

[Do et al., 2018] Do, Q., Martini, B., et Choo, K. R. (2018). **The role of the adversary model in applied security research**. *IACR Cryptol. ePrint Arch.*, 2018:1189.

[docbyte,] docbyte. **Real-time automated anonymization**. https://www.docbyte.com/solutions/anonymization.

[Domingo-Ferrer et al., 2019] Domingo-Ferrer, J., Soria-Comas, J., et Mulero-Vellido, R. (2019). **Steered microaggregation as a unified primitive to anonymize data sets and data streams**. *IEEE Trans. Inf. Forensics Secur.*, 14(12):3298–3311.

[Dong et al., 2011] Dong, W., Zhang, L., Shi, G., et Wu, X. (2011). **Image deblurring and super-resolution by adaptive sparse domain selection and adaptive regularization**. *IEEE Transactions on image processing*, 20(7):1838–1857.

[Dufaux et al., 2010] Dufaux, F., et Ebrahimi, T. (2010). **A framework for the validation of privacy protection solutions in video surveillance**. In *2010 IEEE International Conference on Multimedia and Expo*, pages 66–71.

[Duong et al., 2010] Duong, Q., LeFevre, K., et Wellman, M. P. (2010). **Strategic modeling of information sharing among data privacy attackers**. *Informatica (Slovenia)*, 34(2):151–158.

[Dwork, 2008] Dwork, C. (2008). **Differential privacy: A survey of results**. In *International conference on theory and applications of models of computation*, pages 1–19. Springer.

[Dwork et al., 2006] Dwork, C., McSherry, F., Nissim, K., et Smith, A. (2006). **Calibrating noise to sensitivity in private data analysis**. In *Proceedings of the Third Conference on Theory of Cryptography*, TCC'06, pages 265–284, Berlin, Heidelberg. Springer-Verlag.

[Dwork et al., 2016] Dwork, C., McSherry, F., Nissim, K., et Smith, A. D. (2016). **Calibrating noise to sensitivity in private data analysis**. *J. Priv. Confidentiality*, 7(3):17–51.

[European Parliament, 2018a] European Parliament, C. o. t. E. U. (2018a). **Data minimization**. https://gdpr-info.eu/art-5-gdpr/. Accessed: 2021.

[European Parliament, 2018b] European Parliament, C. o. t. E. U. (2018b). **Gdpr, gerenal data protection regulation**. https://gdpr-info.eu/. Accessed: 2021.

[European Parliament, 2018c] European Parliament, C. o. t. E. U. (2018c). **Personal data**. https://gdpr-info.eu/issues/personal-data/. Accessed: 2021.

[European Parliament, 2019] European Parliament, C. o. t. E. U. (2019). **Data anonymization**. https://ec.europa.eu/eurostat/cros/content/anonymization_en. Accessed: 2021.

[Everingham et al., 2015] Everingham, M., Eslami, S. A., Van Gool, L., Williams, C. K., Winn, J., et Zisserman, A. (2015). **The pascal visual object classes challenge: A retrospective**. *International journal of computer vision*, 111(1):98–136.

[eyedea Regonition, 2021] eyedea Regonition (2021). **Image data anonymization**. https://eyedea.ai/image-data-anonymization/.

[Fayyad et al., 1996] Fayyad, U. M., Piatetsky-Shapiro, G., et Smyth, P. (1996). **Knowledge discovery and data mining: Towards a unifying framework**. In Simoudis, E., Han, J., et Fayyad, U. M., editors, *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining (KDD-96), Portland, Oregon, USA*, pages 82–88. AAAI Press.

[Frome et al., 2009] Frome, A., Cheung, G., Abdulkader, A., Zennaro, M., Wu, B., Bissacco, A., Adam, H., Neven, H., et Vincent, L. (2009). **Large-scale privacy protection in google street view**. In *IEEE 12th International Conference on Computer Vision, ICCV 2009, Kyoto, Japan, September 27 - October 4, 2009*, pages 2373–2380. IEEE Computer Society.

[Garcia, 2016] Garcia, D. (2016). **srez: Adversarial super resolution**. http://github.com/david-gpu/srez.

[Georgevici et al., 2019] Georgevici, A. I., et Terblanche, M. (2019). **Neural networks and deep learning: a brief introduction**.

[Gilchrist, 2016] Gilchrist, A. (2016). **Industry 4.0: the industrial internet of things**. Springer.

[Glorot et al., 2010] Glorot, X., et Bengio, Y. (2010). **Understanding the difficulty of training deep feedforward neural networks**. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pages 249–256. JMLR Workshop and Conference Proceedings.

[Gong et al., 2017] Gong, Q., Luo, J., Yang, M., Ni, W., et Li, X.-B. (2017). **Anonymizing 1:m microdata with high utility**. *Knowledge-Based Systems*, 115(Supplement C):15 – 26.

[Goodfellow et al., 2014] Goodfellow, I. J., Shlens, J., et Szegedy, C. (2014). **Explaining and harnessing adversarial examples**. *arXiv preprint arXiv:1412.6572*.

[Gopalan et al., 2012] Gopalan, R., Taheri, S., Turaga, P., et Chellappa, R. (2012). **A blur-robust descriptor with applications to face recognition**. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(6):1220–1226.

[Greengard, 2015] Greengard, S. (2015). **The internet of things**. MIT press.

[Grira et al., 2004] Grira, N., Crucianu, M., et Boujemaa, N. (2004). **Unsupervised and semi-supervised clustering: a brief survey**. *A review of machine learning techniques for processing multimedia content*, 1:9–16.

[Guo et al., 2013] Guo, K., et Zhang, Q. (2013). **Fast clustering-based anonymization approaches with time constraints for data streams**. *Knowl. Based Syst.*, 46:95–108.

[Hao et al., 2020] Hao, H., Güera, D., Horváth, J., Reibman, A. R., et Delp, E. J. (2020). **Robustness analysis of face obscuration**. In *2020 15th IEEE International Conference on Automatic Face and Gesture Recognition (FG 2020)*, pages 176–183. IEEE.

[Hao et al., 2019] Hao, H., Güera, D., Reibman, A. R., et Delp, E. J. (2019). **Robustness analysis of face obscuration**. *CoRR*, abs/1905.05243.

[Hao et al., 2019] Hao, H., Güera, D., Reibman, A. R., et Delp, E. J. (2019). **A utility-preserving gan for face obscuration.** *Proceedings of the International Conference on Machine Learning, Synthetic Realities: Deep Learning for Detecting Audio Visual Fakes Workshop*.

[Hard et al., 2018] Hard, A., Rao, K., Mathews, R., Beaufays, F., Augenstein, S., Eichner, H., Kiddon, C., et Ramage, D. (2018). **Federated learning for mobile keyboard prediction**. *CoRR*, abs/1811.03604.

[Hartigan et al., 1979] Hartigan, J. A., et Wong, M. A. (1979). **Algorithm as 136: A k-means clustering algorithm**. *Journal of the royal statistical society. series c (applied statistics)*, 28(1):100–108.

[Hatzel, 2021] Hatzel, M. (2021). **Bmw group scaling artificial intelligence for data privacy in production – with innovative anonymisation algorithms**. https://www.press.bmwgroup.com/global/article/detail/T0328838EN/bmw-group-scaling-artificial-intelligence-for-data-privacy-in-production-%E2%80%93-with-innovative-anonymisation-algorithms.

[Haykin, 2010] Haykin, S. (2010). **Neural networks and learning machines, 3/E**. Pearson Education India.

[He et al., 2016a] He, K., Zhang, X., Ren, S., et Sun, J. (2016a). **Deep residual learning for image recognition**. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778.

[He et al., 2016b] He, K., Zhang, X., Ren, S., et Sun, J. (2016b). **Deep residual learning for image recognition**. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pages 770–778. IEEE Computer Society.

[He et al., 2016c] He, K., Zhang, X., Ren, S., et Sun, J. (2016c). **Deep residual learning for image recognition**. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pages 770–778. IEEE Computer Society.

[He et al., 2009] He, Y., et Naughton, J. F. (2009). **Anonymization of set-valued data via top-down, local generalization**. *Proceedings of the VLDB Endowment*, 2(1):934–945.

[Hill et al., 2016a] Hill, S., Zhou, Z., Saul, L., et Shacham, H. (2016a). **On the (in)effectiveness of mosaicing and blurring as tools for document redaction**. *Proceedings on Privacy Enhancing Technologies*, 2016:403 – 417.

[Hill et al., 2016b] Hill, S., Zhou, Z., Saul, L. K., et Shacham, H. (2016b). **On the (in)effectiveness of mosaicing and blurring as tools for document redaction**. *Proc. Priv. Enhancing Technol.*, 2016(4):403–417.

[Hinton et al., 2016] Hinton, G., et Nitsh Srivastava, K. S. (2016). **Neural networks for machine learning, lecture 6a: Overview of mini-batch gradient descent**. https://www.cs.toronto.edu/~hinton/coursera/lecture6/lec6.pdf. Accessed: 2021.

[Hsieh et al., 2020] Hsieh, K., Phanishayee, A., Mutlu, O., et Gibbons, P. B. (2020). **The non-iid data quagmire of decentralized machine learning**. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pages 4387–4398. PMLR.

[Hsu et al., 2019] Hsu, T. H., Qi, H., et Brown, M. (2019). **Measuring the effects of non-identical data distribution for federated visual classification**. *CoRR*, abs/1909.06335.

[Hsu et al., 2020] Hsu, T. H., Qi, H., et Brown, M. (2020). **Federated visual classification with real-world data distribution**. In Vedaldi, A., Bischof, H., Brox, T., et Frahm, J., editors, *Computer Vision - ECCV 2020 - 16th European Conference, Glasgow, UK, August 23-28, 2020, Proceedings, Part X*, volume 12355 of *Lecture Notes in Computer Science*, pages 76–92. Springer.

[Hu et al., 2019] Hu, J., Shen, L., Albanie, S., Sun, G., et Wu, E. (2019). **Squeeze-and-excitation networks**.

[Huang et al., 2017] Huang, G., Liu, Z., Van Der Maaten, L., et Weinberger, K. Q. (2017). **Densely connected convolutional networks**. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4700–4708.

[Huang et al., 2021] Huang, J., Rathod, V., Birodkar, V., Myers, A., Lu, Z., Votel, R., hui Chen, Y., et Chow, D. (2021). **Tensorflow object detection api**. https://github.com/tensorflow/models/tree/master/research/object_detection.

[Intel, 2021] Intel (2021). **Ai-based quality control on every pc for every employee: Bmw group is banking on intel openvino**. https://www.intel.com/content/dam/www/

central-libraries/us/en/documents/2021-06-intel-solution-brief-bmw-robotron-en-final.pdf. Accessed: 2021.

[Jensen et al., 2019] Jensen, M., et Cedric Lauradoux, K. L. (2019). **Pseudonymisation techniques and best practices**. https://www.enisa.europa.eu/publications/pseudonymisation-techniques-and-best-practices. Accessed: 2021.

[Jiao et al., 2019] Jiao, L., Zhang, F., Liu, F., Yang, S., Li, L., Feng, Z., et Qu, R. (2019). **A survey of deep learning-based object detection**. *IEEE access*, 7:128837–128868.

[Jin, 2018] Jin, C. (2018). **Semantic-image-inpainting**. https://github.com/ChengBinJin/semantic-image-inpainting.

[Johnson, 1967] Johnson, S. C. (1967). **Hierarchical clustering schemes**. *Psychometrika*, 32(3):241–254.

[Kaelbling et al., 1996] Kaelbling, L. P., Littman, M. L., et Moore, A. W. (1996). **Reinforcement learning: A survey**. *Journal of artificial intelligence research*, 4:237–285.

[Kairouz et al., 2019] Kairouz, P., McMahan, H. B., Avent, B., Bellet, A., Bennis, M., et et al. (2019). **Advances and open problems in federated learning**. *CoRR*, abs/1912.04977.

[Karami et al., 2017] Karami, E., Shehata, M., et Smith, A. (2017). **Image identification using sift algorithm: performance analysis against different image deformations**. *arXiv preprint arXiv:1710.02728*.

[Keys, 1981] Keys, R. (1981). **Cubic convolution interpolation for digital image processing**. *IEEE transactions on acoustics, speech, and signal processing*, 29(6):1153–1160.

[Khan et al., 2020] Khan, A., Sohail, A., Zahoora, U., et Qureshi, A. S. (2020). **A survey of the recent architectures of deep convolutional neural networks**. *Artificial Intelligence Review*, 53(8):5455–5516.

[Kifer, 2009a] Kifer, D. (2009a). **Attacks on privacy and definetti's theorem**. In *Proceedings of the 2009 ACM SIGMOD International Conference on Management of data*, pages 127–138.

[Kifer, 2009b] Kifer, D. (2009b). **Attacks on privacy and definetti's theorem**. In Çetintemel, U., Zdonik, S. B., Kossmann, D., et Tatbul, N., editors, *Proceedings of the ACM SIGMOD International Conference on Management of Data, SIGMOD 2009, Providence, Rhode Island, USA, June 29 - July 2, 2009*, pages 127–138. ACM.

[Kingma et al., 2014] Kingma, D. P., et Ba, J. (2014). **Adam: A method for stochastic optimization**. *arXiv preprint arXiv:1412.6980*.

[Koh et al., 2021] Koh, J., Lee, J., et Yoon, S. (2021). **Single-image deblurring with neural networks: A comparative survey**. *Computer Vision and Image Understanding*, 203:103134.

[Komkov et al., 2021] Komkov, S., et Petiushko, A. (2021). **Advhat: Real-world adversarial attack on arcface face id system**. In *2020 25th International Conference on Pattern Recognition (ICPR)*, pages 819–826. IEEE.

[Korshunov et al., 2013] Korshunov, P., Melle, A., Dugelay, J.-L., et Ebrahimi, T. (2013). **Framework for objective evaluation of privacy filters**. In SPIE, editor, *Proc. SPIE 8856, Applications of Digital Image Processing XXXVI, 88560T, 26 September 2013, San Diego, California, United States*, San Diego. © 2013 Society of Photo-Optical Instrumentation Engineers.¡br /¿ This paper is published in Proc. SPIE 8856, Applications of Digital Image Processing XXXVI, 88560T, 26 September 2013, San Diego, California, United States and is made available as an electronic preprint with permission of SPIE. One print or electronic copy may be made for personal use only. Systematic or multiple reproduction, distribution to multiple locations via electronic or other means, duplication of any material in this paper for a fee or for commercial purposes, or modification of the content of the paper are prohibited.

[Krizhevsky et al., 2009] Krizhevsky, A., Hinton, G., et others (2009). **Learning multiple layers of features from tiny images**.

[Krizhevsky et al., 2012] Krizhevsky, A., Sutskever, I., et Hinton, G. E. (2012). **Imagenet classification with deep convolutional neural networks**. *Advances in neural information processing systems*, 25:1097–1105.

[Kulkarni et al., 2020] Kulkarni, V., Kulkarni, M., et Pant, A. (2020). **Survey of personalization techniques for federated learning**. *CoRR*, abs/2003.08673.

[Kupyn et al., 2019] Kupyn, O., Martyniuk, T., Wu, J., et Wang, Z. (2019). **Deblurgan-v2: Deblurring (orders-of-magnitude) faster and better**.

[Lander et al., 2001] Lander, K., Bruce, V., et Hill, H. (2001). **Evaluating the effectiveness of pixelation and blurring on masking the identity of familiar faces.** *Applied Cognitive Psychology*, 15(1):101–116.

[LeCun, 1998] LeCun, Y. (1998). **The mnist database of handwritten digits**. *http://yann. lecun. com/exdb/mnist/*.

[LeCun et al., 1998] LeCun, Y., Bottou, L., Bengio, Y., et Haffner, P. (1998). **Gradient-based learning applied to document recognition**. *Proceedings of the IEEE*, 86(11):2278–2324.

[Ledig et al., 2017] Ledig, C., Theis, L., Huszar, F., Caballero, J., Cunningham, A., Acosta, A., Aitken, A. P., Tejani, A., Totz, J., Wang, Z., et Shi, W. (2017). **Photo-realistic single image super-resolution using a generative adversarial network**. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, pages 105–114. IEEE Computer Society.

[LeFevre et al., 2005] LeFevre, K., DeWitt, D. J., et Ramakrishnan, R. (2005). **Incognito: Efficient full-domain k-anonymity**. In *Proceedings of the 2005 ACM SIGMOD international conference on Management of data*, pages 49–60.

[Legislature, 2018] Legislature, C. S. (2018). **Ccpa :california consumer privacy act**. https://oag.ca.gov/privacy/ccpa. Accessed: 2021.

[Li et al., 2007] Li, N., Li, T., et Venkatasubramanian, S. (2007). **t-closeness: Privacy beyond k-anonymity and l-diversity**. In *ICDE*, pages 106–115.

[Li et al., 2008] Li, T., et Li, N. (2008). **Injector: Mining background knowledge for data anonymization**. In *ICDE*, pages 446–455.

[Li et al., 2012] Li, T., Li, N., Zhang, J., et Molloy, I. (2012). **Slicing: A new approach for privacy preserving data publishing**. *IEEE Trans. Knowl. Data Eng.*, 24(3):561–574.

[Li et al., 2017a] Li, Y., Liu, S., Yang, J., et Yang, M. (2017a). **Generative face completion**. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, pages 5892–5900. IEEE Computer Society.

[Li et al., 2017b] Li, Y., Vishwamitra, N., Knijnenburg, B. P., Hu, H., et Caine, K. (2017b). **Effectiveness and users' experience of obfuscation as a privacy-enhancing technology for sharing photos**. *Proc. ACM Hum.-Comput. Interact.*, 1(CSCW).

[Lim et al., 2017] Lim, B., Son, S., Kim, H., Nah, S., et Mu Lee, K. (2017). **Enhanced deep residual networks for single image super-resolution**. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 136–144.

[Lin et al., 2014] Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., et Zitnick, C. L. (2014). **Microsoft coco: Common objects in context**. In *European conference on computer vision*, pages 740–755. Springer.

[Liu et al., 2016] Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y., et Berg, A. C. (2016). **Ssd: Single shot multibox detector**. In *European conference on computer vision*, pages 21–37. Springer.

[Liu et al., 2018] Liu, Y., Ma, S., Aafer, Y., Lee, W., Zhai, J., Wang, W., et Zhang, X. (2018). **Trojaning attack on neural networks**. In *25th Annual Network and Distributed System Security Symposium, NDSS 2018, San Diego, California, USA, February 18-21, 2018*. The Internet Society.

[Liu et al., 2015] Liu, Z., Luo, P., Wang, X., et Tang, X. (2015). **Deep learning face attributes in the wild**. In *2015 IEEE International Conference on Computer Vision, ICCV 2015, Santiago, Chile, December 7-13, 2015*, pages 3730–3738. IEEE Computer Society.

[Luo et al., 2019] Luo, J., Wu, X., Luo, Y., Huang, A., Huang, Y., Liu, Y., et Yang, Q. (2019). **Real-world image datasets for federated learning**. *CoRR*, abs/1910.11089.

[Machanavajjhala et al., 2006] Machanavajjhala, A., Gehrke, J., Kifer, D., et Venkitasubramaniam, M. (2006). **l-diversity: Privacy beyond k-anonymity**. In *Proceedings of the 22nd IEEE International Conference on Data Engineering (ICDE 2006)*, Atlanta Georgia.

[Machanavajjhala et al., 2007] Machanavajjhala, A., Kifer, D., Gehrke, J., et Venkitasubramaniam, M. (2007). **l-diversity: Privacy beyond k-anonymity**. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 1(1):3–es.

[Majeed et al., 2020] Majeed, A., et Lee, S. (2020). **Anonymization techniques for privacy preserving data publishing: A comprehensive survey**. *IEEE Access*.

[Majumdar, 2016] Majumdar, S. (2016). **Image super resolution**. https://github.com/titu1994/Image-Super-Resolution.

[McMahan et al., 2017] McMahan, B., Moore, E., Ramage, D., Hampson, S., et y Arcas, B. A. (2017). **Communication-efficient learning of deep networks from decentralized data**. In Singh, A., et Zhu, X. J., editors, *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics, AISTATS 2017, 20-22 April 2017, Fort Lauderdale, FL, USA*, volume 54 of *Proceedings of Machine Learning Research*, pages 1273–1282. PMLR.

[McPherson et al., 2016] McPherson, R., Shokri, R., et Shmatikov, V. (2016). **Defeating image obfuscation with deep learning.** *CoRR*, abs/1609.00408.

[Mehmood et al., 2016] Mehmood, A., Natgunanathan, I., Xiang, Y., Hua, G., et Guo, S. (2016). **Protection of big data privacy**. *IEEE access*, 4:1821–1834.

[Minaee et al., 2020] Minaee, S., Boykov, Y., Porikli, F., Plaza, A., Kehtarnavaz, N., et Terzopoulos, D. (2020). **Image segmentation using deep learning: A survey**. *CoRR*, abs/2001.05566.

[Minaee et al., 2021] Minaee, S., Boykov, Y. Y., Porikli, F., Plaza, A. J., Kehtarnavaz, N., et Terzopoulos, D. (2021). **Image segmentation using deep learning: A survey**. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

[Mohamed et al., 2016] Mohamed, M. A., Nagi, M. H., et Ghanem, S. M. (2016). **A clustering approach for anonymizing distributed data streams**. In *2016 11th International Conference on Computer Engineering Systems (ICCES)*, pages 9–16.

[Mohammadian et al., 2014] Mohammadian, E., Noferesti, M., et Jalili, R. (2014). **Fast: Fast anonymization of big data streams**. In *Proceedings of the 2014 International Conference on Big Data Science and Computing*, BigDataScience '14, New York, NY, USA. Association for Computing Machinery.

[Muhammad et al., 2015] Muhammad, I., et Yan, Z. (2015). **Supervised machine learning approaches: A survey.** *ICTACT Journal on Soft Computing*, 5(3).

[Nah et al., 2017] Nah, S., Kim, T. H., et Lee, K. M. (2017). **Deep multi-scale convolutional neural network for dynamic scene deblurring**. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[Narayanan et al., 2006] Narayanan, A., et Shmatikov, V. (2006). **How to break anonymity of the netflix prize dataset**. *arXiv preprint cs/0610105*.

[Nasrollahi et al., 2014] Nasrollahi, K., et Moeslund, T. B. (2014). **Super-resolution: a comprehensive survey**. *Machine vision and applications*, 25(6):1423–1468.

[Nawaz et al., 2017] Nawaz, T., Berg, A., Ferryman, J., Ahlberg, J., et Felsberg, M. (2017). **Effective evaluation of privacy protection techniques in visible and thermal imagery**. *Journal of Electronic Imaging*, 26(5):051408.

[Newton et al., 2005a] Newton, E. M., Sweeney, L., et Malin, B. (2005a). **Preserving privacy by de-identifying face images**. *IEEE Transactions on Knowledge and Data Engineering*, 17(2):232–243.

[Newton et al., 2005b] Newton, E. M., Sweeney, L., et Malin, B. (2005b). **Preserving privacy by de-identifying face images**. *IEEE Transactions on Knowledge and Data Engineering*, 17(2):232–243.

[Ng et al., 2014] Ng, H., et Winkler, S. (2014). **A data-driven approach to cleaning large face datasets**. In *2014 IEEE International Conference on Image Processing, ICIP 2014, Paris, France, October 27-30, 2014*, pages 343–347. IEEE.

[Nie et al., 2016] Nie, Y., Huang, L., Li, Z., Wang, S., Zhao, Z., Yang, W., et Lu, X. (2016). **Geospatial streams publish with differential privacy**. In Wang, S., et Zhou, A.,

editors, *Collaborate Computing: Networking, Applications and Worksharing - 12th International Conference, CollaborateCom 2016, Beijing, China, November 10-11, 2016, Proceedings*, volume 201 of *Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering*, pages 152–164. Springer.

[O'Mahony et al., 2019] O'Mahony, N., Campbell, S., Carvalho, A., Harapanahalli, S., Hernandez, G. V., Krpalkova, L., Riordan, D., et Walsh, J. (2019). **Deep learning vs. traditional computer vision**. In *Science and Information Conference*, pages 128–144. Springer.

[Packhäuser et al., 2021] Packhäuser, K., Gündel, S., Münster, N., Syben, C., Christlein, V., et Maier, A. (2021). **Is medical chest x-ray data anonymous?**

[Paszke et al., 2019] Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., et et al. (2019). **Pytorch: An imperative style, high-performance deep learning library**. In Wallach, H. M., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E. B., et Garnett, R., editors, *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 8024–8035.

[Pathak et al., 2016] Pathak, D., Krahenbuhl, P., Donahue, J., Darrell, T., et Efros, A. A. (2016). **Context encoders: Feature learning by inpainting**. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2536–2544.

[Pervaiz et al., 2015] Pervaiz, Z., Ghafoor, A., et Aref, W. G. (2015). **Precision-bounded access control using sliding-window query views for privacy-preserving data streams**. *IEEE Transactions on Knowledge and Data Engineering*, 27(7):1992–2004.

[Punnappurath et al., 2015] Punnappurath, A., Rajagopalan, A. N., Taheri, S., Chellappa, R., et Seetharaman, G. (2015). **Face recognition across non-uniform motion blur, illumination, and pose**. *IEEE Transactions on Image Processing*, 24(7):2067–2082.

[Pushpalwar et al., 2016] Pushpalwar, R. T., et Bhandari, S. H. (2016). **Image inpainting approaches-a review**. In *2016 IEEE 6th International Conference on Advanced Computing (IACC)*, pages 340–345. IEEE.

[Qiu et al., 2016] Qiu, J., Wu, Q., Ding, G., Xu, Y., et Feng, S. (2016). **A survey of machine learning for big data processing**. *EURASIP Journal on Advances in Signal Processing*, 2016(1):1–16.

[Ra et al., 2013] Ra, M.-R., Govindan, R., et Ortega, A. (2013). **P3: Toward privacy-preserving photo sharing**.

[Redmon et al., 2016] Redmon, J., Divvala, S., Girshick, R., et Farhadi, A. (2016). **You only look once: Unified, real-time object detection**. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788.

[Reina et al., 2019] Reina, T., Sheller, M. J., Edwards, B., Martin, J., et Bakas, S. (2019). **Federated learning for medical imaging,ai.intel**. https://www.intel.com/content/www/us/en/artificial-intelligence/posts/federated-learning-for-medical-imaging.html. Accessed: 2021.

[Ren et al., 2015] Ren, S., He, K., Girshick, R., et Sun, J. (2015). **Faster r-cnn: Towards real-time object detection with region proposal networks**. *Advances in neural information processing systems*, 28:91–99.

[Riboni et al., 2012] Riboni, D., Pareschi, L., et Bettini, C. (2012). **Js-reduce: Defending your data from sequential background knowledge attacks**. *IEEE Trans. Dependable Secur. Comput.*, 9(3):387–400.

[Rim et al., 2020] Rim, J., Lee, H., Won, J., et Cho, S. (2020). **Real-world blur dataset for learning and benchmarking deblurring algorithms**. In *European Conference on Computer Vision*, pages 184–201. Springer.

[Ruchaud et al., 2016] Ruchaud, N., et Dugelay, J.-L. (2016). **Automatic face anonymization in visual data: Are we really well protected?** *Electronic Imaging*, 2016(15):1–7.

[Ruder, 2016] Ruder, S. (2016). **An overview of gradient descent optimization algorithms**. *CoRR*, abs/1609.04747.

[Russakovsky et al., 2015a] Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., et others (2015a). **Imagenet large scale visual recognition challenge**. *International journal of computer vision*, 115(3):211–252.

[Russakovsky et al., 2015b] Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M. S., Berg, A. C., et Li, F. (2015b). **Imagenet large scale visual recognition challenge**. *Int. J. Comput. Vis.*, 115(3):211–252.

[Samarati, 2001] Samarati, P. (2001). **Protecting respondents' identities in microdata release**. *IEEE Trans. Knowl. Data Eng.*, 13(6):1010–1027.

[Samarati et al., 1998] Samarati, P., et Sweeney, L. (1998). **Protecting privacy when disclosing information: k-anonymity and its enforcement through generalization and suppression**.

[Schneier, 2015] Schneier, B. (2015). **Data and Goliath: The Hidden Battles to Collect Your Data and Control Your World**. W. W. Norton & Company.

[Schroff et al., 2015] Schroff, F., Kalenichenko, D., et Philbin, J. (2015). **Facenet: A unified embedding for face recognition and clustering**. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 815–823.

[Shafahi et al., 2018] Shafahi, A., Huang, W. R., Najibi, M., Suciu, O., Studer, C., Dumitras, T., et Goldstein, T. (2018). **Poison frogs! targeted clean-label poisoning attacks on neural networks**. *arXiv preprint arXiv:1804.00792*.

[Shan et al., 2020] Shan, S., Wenger, E., Zhang, J., Li, H., Zheng, H., et Zhao, B. Y. (2020). **Fawkes: Protecting privacy against unauthorized deep learning models**. In *29th {USENIX} Security Symposium ({USENIX} Security 20)*, pages 1589–1604.

[Shao et al., 2014] Shao, L., Zhu, F., et Li, X. (2014). **Transfer learning for visual categorization: A survey**. *IEEE transactions on neural networks and learning systems*, 26(5):1019–1034.

[Shen et al., 2018a] Shen, Z., Lai, W., Xu, T., Kautz, J., et Yang, M. (2018a). **Deep semantic face deblurring**. *CoRR*, abs/1803.03345.

[Shen et al., 2018b] Shen, Z., Lai, W., Xu, T., Kautz, J., et Yang, M. (2018b). **Deep semantic face deblurring**. *CoRR*, abs/1803.03345.

[Shorten et al., 2019] Shorten, C., et Khoshgoftaar, T. M. (2019). **A survey on image data augmentation for deep learning**. *Journal of Big Data*, 6(1):1–48.

[sightengine, 2020] sightengine (2020). **Anonymize images: Remove personally identifiable information**. https://sightengine.com/.

[Simonyan et al., 2014] Simonyan, K., et Zisserman, A. (2014). **Very deep convolutional networks for large-scale image recognition**. *arXiv preprint arXiv:1409.1556*.

[Sopaoglu et al., 2020] Sopaoglu, U., et Abul, O. (2020). **A utility based approach for data stream anonymization**. *J. Intell. Inf. Syst.*, 54(3):605–631.

[Srihari,] Srihari, S. N. **Parameter tying and parameter sharing**.

[Sweeney, 2000] Sweeney, L. (2000). **Simple demographics often identify people uniquely**. *Health (San Francisco)*, 671:1–34.

[Sweeney, 2002] Sweeney, L. (2002). **k-anonymity: a model for protecting privacy**. *International Journal on Uncertainty, Fuzziness and Knowledge-based Systems*, 10(5):557–570.

[Tekli et al., 2018] Tekli, J., Al Bouna, B., Bou Issa, Y., Kamradt, M., et Haraty, R. (2018). **(k, l)-clustering for transactional data streams anonymizationa**. In Su, C., et Kikuchi, H., editors, *Information Security Practice and Experience*, pages 544–556, Cham. Springer International Publishing.

[Tekli et al., 2019] Tekli, J., al Bouna, B., Couturier, R., Tekli, G., al Zein, Z., et Kamradt, M. (2019). **A framework for evaluating image obfuscation under deep learning-assisted privacy attacks**. In *17th International Conference on Privacy, Security and Trust, PST 2019, Fredericton, NB, Canada, August 26-28, 2019*, pages 1–10. IEEE.

[Tekli et al., 2021] Tekli, J., Aoun, G., Charbel, A., Dib, F., et Anwar, R. (2021). **Bmw-anonymization-api**. https://github.com/BMW-InnovationLab/BMW-Anonymization-API.

[Tekli et al., NDa] Tekli, J., Bouna, B. A., Tekli, G., Charbel, A., et Couturier, R. (N.D.a). **Leveraging deep learning-assisted attacks against image obfuscation via federated learning**. Under Review.

[Tekli et al., NDb] Tekli, J., Bouna, B. A., Tekli, G., Couturier, R., et Kamradt, M. (N.D.b). **A framework for evaluating image obfuscation under deep learning-assisted privacy attacks**. Under Review.

[Terrovitis et al., 2012] Terrovitis, M., Mamoulis, N., Liagouris, J., et Skiadopoulos, S. (2012). **Privacy preservation by disassociation**. *Proc. VLDB Endow.*, 5(10):944–955.

[Turk et al., 1991] Turk, M. A., et Pentland, A. P. (1991). **Face recognition using eigenfaces**. In *Proceedings. 1991 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 586–591.

[vaisala, 2018] vaisala (2018). **Automated anonymization of visuals with computer vision**. https://www.vaisala.com/en/blog/2020-10/automated-anonymization-visuals-computer-vision.

[Verbraeken et al., 2020a] Verbraeken, J., Wolting, M., Katzy, J., Kloppenburg, J., Verbelen, T., et Rellermeyer, J. S. (2020a). **A survey on distributed machine learning**. *ACM Computing Surveys (CSUR)*, 53(2):1–33.

[Verbraeken et al., 2020b] Verbraeken, J., Wolting, M., Katzy, J., Kloppenburg, J., Verbelen, T., et Rellermeyer, J. S. (2020b). **A survey on distributed machine learning**. *ACM Comput. Surv.*, 53(2):30:1–30:33.

[Viola et al., 2001] Viola, P., et Jones, M. (2001). **Rapid object detection using a boosted cascade of simple features**. In *Proceedings of the 2001 IEEE computer*

society conference on computer vision and pattern recognition. CVPR 2001, volume 1, pages I–I. Ieee.

[Wang et al., 2018] Wang, J., Deng, C., et Li, X. (2018). **Two privacy-preserving approaches for publishing transactional data streams**. *IEEE Access*, pages 1–1.

[Wang et al., 2010] Wang, P., Zhao, L., Lu, J., et Yang, J. (2010). **Sanatomy: Privacy preserving publishing of data streams via anatomy**. In *2010 Third International Symposium on Information Processing*, pages 54–57.

[Wang et al., 2017] Wang, X., Peng, Y., Lu, L., Lu, Z., Bagheri, M., et Summers, R. M. (2017). **Chestx-ray8: Hospital-scale chest x-ray database and benchmarks on weakly-supervised classification and localization of common thorax diseases**. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[Weiss et al., 2016] Weiss, K., Khoshgoftaar, T. M., et Wang, D. (2016). **A survey of transfer learning**. *Journal of Big data*, 3(1):1–40.

[Wong et al., 2011a] Wong, R. C.-W., Fu, A. W.-C., Wang, K., Yu, P. S., et Pei, J. (2011a). **Can the utility of anonymized data be used for privacy breaches?** *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 5(3):1–24.

[Wong et al., 2011b] Wong, R. C.-W., Fu, A. W.-C., Wang, K., Yu, P. S., et Pei, J. (2011b). **Can the utility of anonymized data be used for privacy breaches?** *ACM Trans. Knowl. Discov. Data*, 5(3):16:1–16:24.

[Wu et al., 2020a] Wu, Q., He, K., et Chen, X. (2020a). **Personalized federated learning for intelligent iot applications: A cloud-edge based framework**. *IEEE Open J. Comput. Soc.*, 1:35–44.

[Wu et al., 2020b] Wu, Z., Wang, H., Wang, Z., Jin, H., et Wang, Z. (2020b). **Privacy-preserving deep action recognition: An adversarial learning framework and a new dataset**. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

[Xiao et al., 2006a] Xiao, X., et Tao, Y. (2006a). **Anatomy: Simple and effective privacy preservation**. In *Proceedings of 32nd International Conference on Very Large Data Bases (VLDB 2006)*, Seoul, Korea.

[Xiao et al., 2006b] Xiao, X., et Tao, Y. (2006b). **Anatomy: Simple and effective privacy preservation**. In *Proceedings of the 32nd international conference on Very large data bases*, pages 139–150.

[Xie et al., 2017] Xie, S., Girshick, R., Dollár, P., Tu, Z., et He, K. (2017). **Aggregated residual transformations for deep neural networks**.

[Xu, 2008] Xu, S. (2008). **Collaborative attack vs. collaborative defense**. In Bertino, E., et Joshi, J. B. D., editors, *Collaborative Computing: Networking, Applications and Worksharing, 4th International Conference, CollaborateCom 2008, Orlando, FL, USA, November 13-16, 2008, Revised Selected Papers*, volume 10 of *Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering*, pages 217–228. Springer / ICST.

[Yang et al., 2021] Yang, K., Yau, J., Fei-Fei, L., Deng, J., et Russakovsky, O. (2021). **A study of face obfuscation in imagenet**.

[Yang et al., 2019a] Yang, Q., Liu, Y., Cheng, Y., Kang, Y., Chen, T., et Yu, H. (2019a). **Federated Learning**. Morgan & Claypool Publishers.

[Yang et al., 2019b] Yang, W., Zhang, X., Tian, Y., Wang, W., Xue, J., et Liao, Q. (2019b). **Deep learning for single image super-resolution: A brief review**. *IEEE Trans. Multim.*, 21(12):3106–3121.

[Yeh et al., 2017] Yeh, R. A., Chen, C., Lim, T., Schwing, A. G., Hasegawa-Johnson, M., et Do, M. N. (2017). **Semantic image inpainting with deep generative models**. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, pages 6882–6890. IEEE Computer Society.

[Yu et al., 2020] Yu, T., Bagdasaryan, E., et Shmatikov, V. (2020). **Salvaging federated learning by local adaptation**. *CoRR*, abs/2002.04758.

[Yu et al., 2016] Yu, X., et Porikli, F. (2016). **Ultra-resolving face images by discriminative generative networks**. In Leibe, B., Matas, J., Sebe, N., et Welling, M., editors, *Computer Vision - ECCV 2016 - 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part V*, volume 9909 of *Lecture Notes in Computer Science*, pages 318–333. Springer.

[Zakerzadeh et al., 2011] Zakerzadeh, H., et Osborn, S. L. (2011). **Faanst: Fast anonymizing algorithm for numerical streaming data**. In Garcia-Alfaro, J., Navarro-Arribas, G., Cavalli, A., et Leneutre, J., editors, *Data Privacy Management and Autonomous Spontaneous Security*, pages 36–50, Berlin, Heidelberg. Springer Berlin Heidelberg.

[Zakerzadeh et al., 2013] Zakerzadeh, H., et Osborn, S. L. (2013). **Delay-sensitive approaches for anonymizing numerical streaming data**. *Int. J. Inf. Sec.*, 12(5):423–437.

[Zhang et al., 2017] Zhang, J., Li, H., Liu, X., Luo, Y., Chen, F., Wang, H., et Chang, L. (2017). **On efficient and robust anonymization for privacy protection on massive streaming categorical information**. *IEEE Trans. Dependable Secur. Comput.*, 14(5):507–520.

[Zhang et al., 2020] Zhang, K., Gool, L. V., et Timofte, R. (2020). **Deep unfolding network for image super-resolution**.

[Zhao et al., 2017] Zhao, H., Shi, J., Qi, X., Wang, X., et Jia, J. (2017). **Pyramid scene parsing network**. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2881–2890.

[Zhao et al., 2018] Zhao, Y., Li, M., Lai, L., Suda, N., Civin, D., et Chandra, V. (2018). **Federated learning with non-iid data**. *CoRR*, abs/1806.00582.

[Zhou et al., 2017] Zhou, B., Zhao, H., Puig, X., Fidler, S., Barriuso, A., et Torralba, A. (2017). **Scene parsing through ade20k dataset**. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 633–641.

[Zhou Wang et al., 2004] Zhou Wang, Bovik, A. C., Sheikh, H. R., et Simoncelli, E. P. (2004). **Image quality assessment: from error visibility to structural similarity**. *IEEE Transactions on Image Processing*, 13(4):600–612.

[Zhu et al., 2017] Zhu, J.-Y., Park, T., Isola, P., et Efros, A. A. (2017). **Unpaired image-to-image translation using cycle-consistent adversarial networks**. In *Proceedings of the IEEE international conference on computer vision*, pages 2223–2232.

List of Figures

List of Tables

List of Definitions

Document generated with LaTeX and:
the LaTeX style for PhD Thesis created by S. Galland — http://www.multiagent.fr/ThesisStyle
the tex-upmethodology package suite — http://www.arakhne.org/tex-upmethodology/

école doctorale **sciences pour l'ingénieur et microtechniques**

Title: Designing and evaluating anonymization techniques for images and relational data streams via Machine Learning approaches at BMW Group

Keywords: Data Privacy, Anonymization , Face image Obfucsation, Deep Learning, Deep Learning-assisted attacks, Federated Learning, Collaborative Attacks, Data Stream, Correlation

Abstract:

Individual's privacy and anonymity is becoming highly critical in our data-driven world due to the vast amount of data being generated and processed daily (e.g., Industry 4.0). Data anonymization is the process of creating anonymous information, namely information which does not relate to an identified or identifiable natural person in such a manner that the data subject is not or no longer identifiable. Privacy regulations compel data-driven companies to guarantee a level of anonymization that requires "irreversibility preventing identification of the data subject", taking into account all the means "reasonably likely to be used" for identification. Therefore, we (i) propose and implement several anonymization techniques and tools in the context of images and relational data streams and (ii) assess the robustness of these techniques by simulating adversaries with different knowledge and several attacking capabilities. In the first contribution, we design and implement an anonymization tool that localizes identifying/sensitive features in images/videos via Deep Learning DL-based localization techniques (i.e., semantic segmentation) and obfuscates it accordingly via pixelating, blurring, or masking. In the second contribution, we propose a recommendation framework that evaluates the robustness of image obfuscation techniques and recommends the most resilient obfuscation against adversaries executing DL-assisted attacks (e.g., *restoration*-based or *recognition*-based attacks). In addition, three threat levels are studied thoroughly based on the adversary's knowledge (e.g., background knowledge). In the third contribution, we empirically demonstrate how adversaries can remedy their lack of knowledge and leverage their attacking capabilities, against obfuscated facial images, by collaborating via Federated Learning. Seven collective threat levels are defined and studied based on the background knowledge of the adversaries and the sharing of their knowledge. Finally, we address in the fourth contribution the correlation problem in the anonymization of a transactional relational data stream. A bucketization-based technique, entitled (k,l)-clustering, is proposed to prevent such privacy breaches by ensuring that the same k individuals remain grouped together over the entire anonymized stream.

Titre : Conception et évaluation de techniques d'anonymisation des images et des flux de données relationnels via des approches d'apprentissage automatique à BMW Group.

Mots-clés : Confidentialité des données, Anonymisation, Obfuscation des images, Apprentissage profond, Attaques assistées par l'apprentissage profond, Apprentissage fédéré, Attaques collaboratives, flux des données, Corrélation

Résumé :

La protection des données à caractère personnel est essentielle et vitale dans notre monde axé sur les donnes (e.g. industrie 4.0). L'anonymisation est un processus qui modifie les données de telle manière que la personne concernée ne soit pas ou plus identifiable. Les réglementations de protection des données obligent souvent les entreprises qui utilisent des données de garantir un niveau d'anonymisation qui exige "l'irréversibilité empêchant la ré-identification de la personne concernée", en tenant compte de tous les moyens "raisonnablement susceptibles d'être utilisés" pour l'identification. Par conséquent, nous (i) proposons et implémentons plusieurs techniques et outils d'anonymisation dans le contexte des images et des flux de données relationnels et (ii) évaluons la robustesse de ces techniques en simulant des adversaires avec plusieurs capacités d'attaque. Dans la première contribution, nous concevons masquage. Dans la deuxième contribution, nous proposons un « Framework » qui évalue la robustesse des techniques d'anonymisation des images et recommande la technique la plus résiliente contre des adversaires qui exécutent des attaques assistées par DL (par exemple, des attaques qui reconstruisent/reconnaissent les pixels anonymes). En outre, nous étudions trois niveaux d'attaque dont chacun dépend des connaissances « knowledge » de l'adversaire à propos des images anonymisés. Dans la troisième contribution, nous démontrons d'une manière empirique comment les adversaires peuvent remédier à leur manque de connaissances et améliorer leurs capacités d'attaque, contre des images anonymisés, en collaborant via « Federated Learning ». Nous définissons sept niveaux d'attaque collective en fonction des connaissances des adversaires et du partage de leurs connaissances. Dans la quatrième

UBFC UNIVERSITÉ BOURGOGNE FRANCHE-COMTÉ
Université Bourgogne Franche-Comté
32, avenue de l'Observatoire
25000 Besancon, France