Acta Cybernetica **25** (2021) 233–255.

A Comparative Study on the Privacy Risks of Face Recognition Libraries*

István Fábián[ab] and Gábor György Gulyás[ac]

Abstract

The rapid development of machine learning and the decreasing costs of computational resources has led to a widespread usage of face recognition. While this technology offers numerous benefits, it also poses new risks. We consider risks related to the processing of face embeddings, which are floating point vectors representing the human face. Previously, we showed that even simple machine learning models are capable of inferring demographic attributes from embeddings, leading to the possibility of re-identification attacks. This paper proposes a new data protection evaluation framework for face recognition, and examines three popular Python libraries for face recognition (OpenCV, Dlib, InsightFace), comparing their face detection performance and inspecting how much risk each library's embeddings pose regarding the aforementioned data leakage. Our experiments were conducted on a balanced face image dataset of different sexes and races, allowing us to discover biases. Based on our results, Dlib has a significant FNR of 4.2% on the total dataset, and an eccentric 5.9% FNR on black people. Finally, our findings indicate that all three libraries could enable sex or race based discrimination in re-identification attacks.

Keywords: face recognition, machine learning, privacy

1 Introduction

With the trend of technology getting cheaper and the advance of smart technologies, security and surveillance cameras are getting more and more widespread recently.

*The research has been supported by the European Union, co-financed by the European Social Fund (EFOP-3.6.2-16-2017-00013, Thematic Fundamental Research Collaborations Grounding Innovation in Informatics and Infocommunications). Project no. FIEK_16-1-2016-0007 has been implemented with the support provided from the National Research, Development and Innovation Fund of Hungary, financed under the Centre for Higher Education and Industrial Cooperation - Research infrastructure development (FIEK_16) funding scheme.

[a]Department of Automation and Applied Informatics, Faculty of Electrical Engineering and Informatics, Budapest University of Technology and Economics, Hungary

[b]E-mail: fabian@aut.bme.hu, ORCID: 0000-0003-0293-0335

[c]E-mail: gabor.gulyas@aut.bme.hu, ORCID: 0000-0003-0877-0088

DOI: 10.14232/actacyb.289662

According to recent news, Chongqing, a single Chinese city alone has more than 2.5 million surveillance cameras installed [19]. This problem set is not constrained to countries similar to China, as for example London also has more than 600,000 of such cameras [19]. These devices enable emerging artificial intelligence based face recognition technologies in the physical world at scale. This will certainly have a significant impact on the society as a whole, and on the personal level as well, as these advances enable surveillance at large extent as never seen before.

In this paper, we look at the case of the large-scale storing and processing of face imprints generated by face recognition technologies. This technology uses the photo or a video frame containing a person's face to extract an imprint from it. The imprint, or the embedding, describes the face based on its unique characteristics, thus it can be used for identification. When generated by deep learning techniques, the embedding is usually hard for a human to interpret, as usually it is a vector of real values. The length of this vector may vary depending on the used technique.

Identification (i.e. the recognition) works by comparing multiple embedding vectors to each other by calculating similarity between them (e.g. via the Euclidean or Manhattan distance). At the end, pairwise similarities of the embeddings indicate whether the two faces should be considered to be of the same person. It is presumed that the lower the distance, the higher the similarity, and the similarity of embeddings is proportional to the similarity of the faces. Usually if the distance is below a certain threshold, the embeddings are considered to belong to the same person. Or in other words, identification is effectively done by clustering embeddings.

In our research, we are concerned with the possible privacy risks related to utilizing face recognition embeddings. This paper extends our previous work, "On the Privacy Risks of Large-Scale Processing of Face Imprints" [11].

In our previous work we have evaluated a re-identification attack scheme through where we simulated the attacker precision in predicting demographics from embeddings (without executing any machine learning tasks). In our current work, we look in deeper details into these attacks.

We provide a thorough comparison of three popular face recognition Python libraries: OpenCV, Dlib, and InsightFace. We compare these libraries from two different perspectives on people of both sexes, four different races and multiple age groups. First, we consider the face detection performance of these libraries. Then, we consider embedding inference, where we examine how accurately we can train a machine learning model to infer demographic data from the embeddings generated by the libraries.

We also build on results from our previous work in "De-anonymizing Facial Recognition Embeddings"[12] where we showed that re-identification attacks by inferring demographic data from face embeddings are a valid threat (see Figure 1), which justifies the relevance of our current research.

We consider the following setup: cameras observe some areas (for example at a company, or in a public space) and extract facial embeddings of people passing by. Either the cameras themselves are capable of doing the extraction, or they transfer their footage to a capable server device that would do so. Depending on the use case (tracking, authentication, identification, etc.), either embeddings are stored in

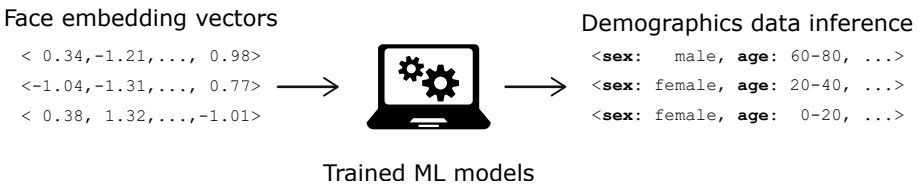


Figure 1: A possible privacy concern regarding face recognition is the inference of sensitive demographic data from face embeddings through inference of specific machine learning models.

a database to be used later on, or are compared in real-time to other embeddings that are already stored in the database.

The reason why the processing may be concerning is that embeddings are considered biometric data and unlike other biometric data such as fingerprints, facial images can be easily captured without a person's knowledge and consent, and also at a large scale [2]. Therefore, in this paper, we look at risks related to the processing of embeddings, more specifically we analyze the privacy risk of demographic-based person re-identification by using face imprints.

This paper is structured as follows. Section 2 summarizes relevant research related to this topic, including how face recognition works and what its privacy concerns are. Section 3 introduces a proposed new data protection evaluation framework for face recognition. Section 4 demonstrates a theoretical attack and evaluates its results. Section 5 compares three popular face recognition libraries and introduces the dataset on which they were tested. Finally, Section 6 concludes the paper with a summary of its main takeaways.

2 Related work

In this section we review facial recognition: its history, how it started, major breakthroughs, and how deep learning based state of the art face recognition systems work. Then, we introduce and discuss the most important features of the three Python libraries we used in our work. At the end of the section we also discuss ethical and privacy concerns related to the application of facial recognition.

2.1 About face recognition

Historically, the dawn of facial recognition began in the 1960s, when researchers began to use computers to recognize human faces [30]. The first trial was a man-machine approach, where human personnel had to manually mark facial landmarks on photographs (e.g. eyes, eyebrows, ears, nose, lips), and the coordinates of these landmarks were then transformed by a computer to undo the effects of variations in head rotation and tilt. Then for each person a list of these coordinates were stored, and in the recognition phase, the distances were calculated between the

photograph and all the stored records, and the lowest distance was supposed to reveal the recognized person.

The next major milestone was reached in the 1980s and 1990s, when researchers came up with the eigenfaces approach [31]. The goal of this approach was to be able to represent faces as 1 dimensional vectors (instead of 3 dimensional RGB images), as a combination of predetermined "base" faces, called eigenfaces. The basic idea was to take a facial image dataset, align and center all faces, and create a data matrix by turning the images into vectors. This was followed by calculating the mean face (μ) by averaging the data matrix. The eigenfaces (e) were then constructed by determining the matrix's eigenvectors and reshaping them into images. Afterwards, each new face X could be represented as the mean face plus a linear combination of the eigenfaces: $X = \mu + w_1 \cdot e_1 + w_2 \cdot e_2 + ... + w_n \cdot e_n$, where w_i represents the coefficients of the eigenfaces. In the recognition phase, the similarities between different faces could be determined by calculating a distance (e.g. Euclidean distance) between the coefficients of the eigenfaces belonging to different individuals (where a lower distance meant closer similarity). The biggest advantage of this approach was that it no longer required human manual input, and it was completely automated so it worked even in real-time settings. However, a significant drawback was that it was very sensitive to lightning, scale and facial expression variations, so it could only work in highly controlled environments.

The next breakthrough, which is the current state of the art in face recognition, was made possible by the utilization of deep learning algorithms. These algorithms take the pixels of a photo (or frame) of a person as an input and firstly detect the face in the image. Various techniques can be used for face detection, such as Histogram of Oriented Gradients (HOG) [4][25], Haar-Cascades [32] or even a neural network. Once the face is detected, certain transformations are performed to make it frontal facing and centered, and finally a vector of floating point numbers is generated as an output. These vectors are supposed to describe the human face's unique features.

To create such vectors, a special training setup is needed. Most often, a Siamese network architecture [33] and a special loss function, such as triplet loss [26] is used, where during each iteration of the training three identical networks (hence the name "Siamese" networks) are fed three different face images, two of the same person (the "anchor" and the "positive" image) and one of a different person (the "negative image"). The goal of the training is to modify the weights of the network such that the output embeddings of the anchor and positive images will be close in vector space, while the negative image's embedding will be farther. The advantage of this training setup is that the network can learn to generalize and cluster the same faces together without having to see each possible human face during training.

Then, during recognition phase, these output vectors, also known as face embedding vectors, are compared according to a certain distance metric (e.g. the Euclidean or Manhattan distance) to determine whether two embeddings belong to the same face or not. The length of this vector may differ from implementation to implementation, for example some libraries might generate a 128 dimensional vector [26][21], whereas other libraries generate a 512 dimensional vector [8].

2.2 Face recognition libraries

The three libraries we used in our work are as follows.

The OpenCV library [1] implements a deep convolutional neural network based on the FaceNet [26] structure. Previous networks were trained on a set of known identities and used an intermediate bottleneck-layer to learn a generalized representation of faces for recognition. This setting was inefficient and problematic, because the bottleneck layer couldn't always generalize to new faces, and the representation size of faces were usually thousands of dimensions large. In contrast, FaceNet is an end-to-end solution that directly maps images of faces into the 128-dimensional embedding metric space without requiring a representational bottleneck-layer, using the triplet-based loss function described above. FaceNet was built on two different architectures, the Zeiler Fergus and the GoogLeNet style Inception models. While the Zeiler Fergus model has 140 million parameters, the Inception model has only 7.5 million, making its usage possible on lower computation capacity devices, such as mobile phones.

The Dlib library [21] is based on a ResNet-34 [15] structure deep convolutional neural network. In theory, by increasing the network depth, performance should improve as the model should be able to learn more features. In practice there are, however, obstacles to increasing the depth indefinitely. One obstacle is the problem of the vanishing/exploding gradients, which can be solved by normalized initialization and batch normalization. Another obstacle is that researchers found that adding more layers to a network could actually result in higher training error. The key idea of residual networks such as ResNet-34 is the addition of residual layers to deep convolutional nets. In these models, shortcut connections are added that skip certain layers, performing identity mapping between two non-neighboring layers, thereby not only solving the problem of higher training errors, but actually producing accuracy gains in very deep networks.

The InsightFace [8] library also utilizes a deep convolutional neural network which was based on multiple other networks (ResNet, MobilefaceNet[3], Inception-ResNet_v2 [29], DenseNet [17], etc.) and besides triplet (Euclidean/Angular) loss it also uses multiple loss functions including Additive Angular Margin Loss (ArcFace), which was created with the specific aim to obtain highly discriminative features for face recognition [7]. By maximizing face class separability (i.e. clustering faces belonging to the same person much more closely than other loss functions), this approach enables the network to be less sensitive towards pose and age variations.

The performance of FR libraries is usually tested by benchmarking them on various face image datasets, including the Labeled Faces in the Wild dataset [18], which is the most common benchmarking dataset. From this perspective, Dlib achieves 99.38%, OpenCV achieves 99.63% and InsightFace achieves the highest 99.83% accuracy on this dataset.

2.3 Ethical concerns and privacy risks

While FR technology offers a lot of benefits to humanity and it already has a lot of uses in our everyday lives (e.g. smartphones unlocking by recognizing their owner's face, automatic tagging of people on social networking sites, automated border control gates, finding a lost person, tracking someone etc.) this technology could also pose numerous threats to society.

One of the biggest concerns is that of discrimination. It could be caused not only by face recognition itself, but also by the underlying face detection technology. Some face detection algorithms (like the previously mentioned Haar-Cascades) work by detecting edges, lines and shapes in images. Under certain circumstances (e.g. poor lightning conditions), these techniques work better on light skinned individuals, and perform worse on darker skinned people. A good example of this was when Hewlett-Packard's motion-tracking webcams failed to detect a black person's face [27], but Google Photos also struggled with detecting black persons, mislabeling them for gorillas instead [34]

To analyze the level of discrimination, the Face Recognition Vendor Test conducted by the National Institute of Standards and Technology (NIST) examined the accuracy variations and potential biases across different demographic groups based on sex, age and race [14]. In their study, they examined the performance of 189 face recognition algorithms made by 99 different developers, on over 18 million photographs taken of more than 8 million people. Their report examined the variation between false positives and false negatives for the different demographics analyzed. Overall they found that false positives were much more common than false negatives, and the ratio of false positives was higher among West- and East-African, East-Asian, American-Indian and African-American groups. They also found the false positive ratio to be higher among women, and the youngest and oldest individuals. Considering some of the use cases (e.g. law enforcement usage to identify suspects) these high false positive rates could have a lot of negative consequences on people's lives, like in the case of 3 black men who were mistakenly identified and falsely arrested [16]. Knowing about the existence of these sex and race dependent face recognition performance variations, in our work we examined whether similar demographic biases are also present in the inference of sensitive details from the embeddings. Our results are discussed in Section 5.

Apart discrimination and bias issues, face recognition also poses privacy threats. According to the General Data Protection Regulation (GDPR), face embeddings are biometric data, as the GDPR defines biometric data as *"personal data resulting from specific technical processing relating to the physical, physiological or behavioural characteristics of a natural person, which allow or confirm the unique identification of that natural person, such as facial images or dactyloscopic data"* [10]. As such, processing face embeddings are forbidden by default, and their processing requires special conditions to be met or to have all concerned subject to consent. However, by the nature of video surveillance, consent can be very difficult to obtain; as in public spaces data subjects may not even be aware of being surveilled. Another problematic aspect of processing biometric data is that while it can be in fact used

for identification, it should not be used for authentication. Unlike a password, a person's biometric traits are not replaceable and not revocable, which may lead to severe security risks (e.g. biometric data leakage in database hacks). For these reasons, face recognition should be used as a second factor authentication at most, which is not always the case in real world applications.

Privacy threats arise in different shapes and colors in different sectors. By governments in the public sector, there could be misguided use cases that could even threaten democracy as we know it (e.g. mass surveillance using FR in totalitarian regimes, law enforcement usages discriminating certain groups). Risks concerning individuals relate to using FR services on cloud providers that may not respect or protect their data carefully (e.g. Facebook automatic facial recognition on uploaded images posing interdependent privacy risks). In the private sector there may be irresponsible use cases where the nature of biometric face embeddings is not treated with enough caution (e.g. face image or face embedding database leaks, face spoofing attacks, leaking sensitive information via face embeddings, etc).

Due to the numerous privacy harms that could result from the irresponsible usage of facial recognition, in the following Section we introduce a novel data protection evaluation framework that can be used to examine the potential risks in a systematic way.

3 Facial Recognition Data Protection Impact Assessment Framework

In this Section we propose a detailed data protection evaluation framework for facial recognition. Such framework could be a helpful guide in conducting the Data Protection Impact Assessment (DPIA) for applications that utilize face recognition.

Under the GDPR, it could be a mandatory requirement to conduct a DPIA for any case where sensitive data might be published or leaked (e.g. biometric data such as face imprints) [10]. It necessitates the data processor to examine the privacy harms resulting from a potential attack, and to make certain technical and organizational measures so as to minimize the impact of such an attack. As part of the DPIA, the data handler has to evaluate all plausible settings and risk scenarios, so conducting the DPIA is non-linear, cyclic task.

Our proposed framework enables a systematic approach to conduct the DPIA according to GDPR guidelines. To the best of our knowledge, currently no such framework exists specifically for face recognition related data processing. The framework is seen in Figure 2.

The first stage represents the processed data by the data processor. In our case, the data includes face embeddings along with some extra information. This may include sensitive, directly or non-directly identifying personal information for individuals, depending on the concrete use case. (One example could be a camera system at an airport, that could record the embeddings of people entering a prayer room, posing the risk of sensitive information leakage.)

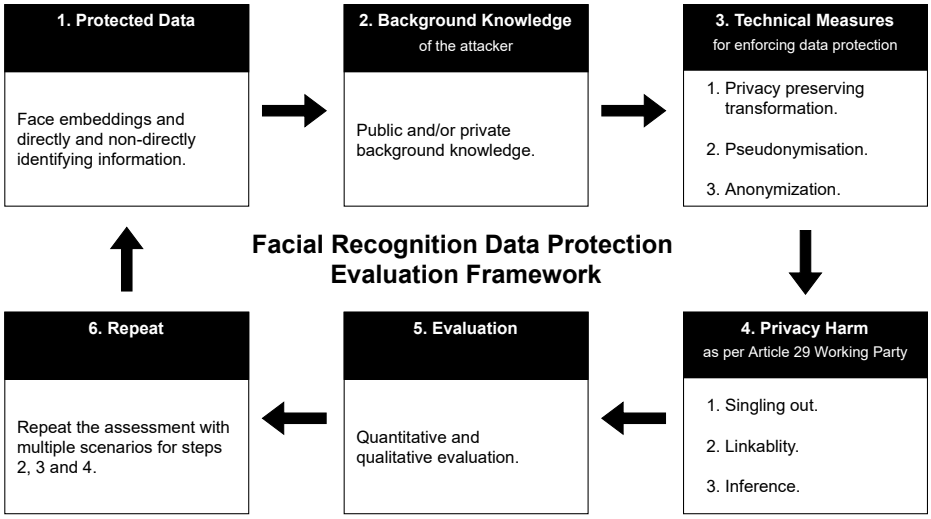


Figure 2: Our proposed framework for helping to carry out the Data Protection Impact Assessment of facial recognition, as required by the GDPR.

The second stage illustrates the potential background knowledge of the attacker. Depending on the nature of the attack, the attacker might have access to only public or both public and private auxiliary information that she could use for an attack. In case of an outside attack, the hacker could only use publicly available data to carry out a privacy attack (e.g. social media posts and photos, voter registration lists, etc.). However, in case of an inside attack, the malicious actor could have access to protected data that has high overlap with the published or leaked original dataset, thus presents higher risk (e.g. if the attacker is a system administrator).

The third stage details the technical measures that could be taken by the data processor to minimize the privacy harms resulting from attacks. The data processor could apply privacy preserving transformations (e.g. mapping, hashing, data perturbations), pseudonymization (e.g. cryptographic or hashing techniques) and/or anonymization (e.g. k-anonymity) in this step. The point of these measures is to narrow down the possibilities of a malicious party to minimize the impacts of the attack.

The fourth stage discusses the potential privacy attacks by the malicious party as per the GDPR. The Article 29 Working Party determined three different attack types [13]: singling out (the malicious actor successfully identifies an individual in the dataset), linkability (connecting two records of the same individual from different databases) and inference (finding out new information about individuals with high probability).

The fifth stage distinguishes two approaches for evaluating the impact of an attack: quantitative and qualitative approaches. Quantitative approaches take

into account the success rate of an attacker, such as the percentage of individuals re-identified, true positive rate, false positive rate, recall and other similar metrics. On the other hand, qualitative approaches deal with the nature of the suffered privacy harm, such as the leakage of sensitive information like sexual, political, religious orientation, behavioral preferences or the revelation of someone's location. These could have moral or material impact on the degree of personal freedom of individuals.

Finally, the sixth stage emphasises the cyclic nature of the DPIA. Namely, the quantitative and qualitative evaluation of the attack must be completed for multiple different scenarios for the assumed background knowledge, technical measures taken, and privacy harm considered.

We believe that the above introduced general framework is a helpful starting point for preparing the DPIA and to analyze numerous different privacy threats. In our work, we considered inference based linkability as the privacy harm, where the attacker uses her background knowledge combined with demographic information inferred from the embeddings to carry out a re-identification attack. The following Section details our work regarding the attack and the estimated risk.

4 Attack and risk level estimation

Previously, we have shown that sex, race and age can be predicted with high accuracy from face embedding vectors [12], but researchers showed that even the original face image can be reconstructed from the embeddings [22], which means that certain types of data that can be determined by looking at a person's face, such as hair color, glasses, etc., are also stored in embeddings. Such traits can be referred to as soft biometric traits [5], which define some information about an individual, but are not distinctive enough to make them uniquely identifiable.

The problem is that personal attributes that are not personally identifiable information yet can be combined together or indirectly merged with external data sources in order to put back the names over de-identified data (i.e. where all directly identifying attributes are removed) [28]. We call such procedures re-identification attacks. Consider an example where a company publishes a database with information about its employees, de-identified by removing explicitly identifying fields (names, email, etc.) and replacing them with unique random IDs. While this database alone might be considered de-identified, but an attacker may link records from this company-related dataset to a medical dataset's corresponding records by using demographic data.

There are several ways how an attacker can be successful at re-identification by using face embeddings:

- By matching embeddings: e.g. the attacker has a photo, extracts an embedding and looks for a match in a database containing embeddings. As mentioned in Section 1, if the distance between the two embeddings is below a threshold then the embeddings belong to the same person with some probability.

- If direct search of embeddings is not possible, the attacker could reconstruct the face from the embeddings in the database [22], and run a visual search in a face database (e.g. photos on a social network).
- Knowing that embeddings contain demographic data about the data subject, the attacker can try reconstructing such data from the stored embedding itself (e.g. using a machine learning model trained for this task) and using that to do cross matching in another database.

As we know that demographic data predictions are feasible, we consider the third class of attacks, which is an inference based linkability attack as per our proposed framework. This is also motivated with the fact that the zip code, sexuality and date of birth combined together provide a unique identifier for 87% of the population based on US census data. [28] Referring back to our framework, if an attacker combines her background knowledge with accurately predicted demographic data from embeddings, and knows further pieces of background information such as place of work or residence, she will be able to look up the identity of the data subject by looking her or him up on social network sites (e.g. on LinkedIn).

Let us explain this concrete attack as follows (see Figure 3). Let us assume a company where the employees are monitored by FR-capable smart CCTVs that store the extracted face embeddings in a central database. If the attacker manages to get the database, she can perform the following attack. In the 1st step the attacker downloads a publicly available face images dataset. In the 2nd step, the attacker labels the downloaded face images with demographic attributes such as sex, race and age (if they are not already labeled by default) and runs FR on them to extract the face embeddings. Afterwards, she trains a machine learning model to classify embeddings into demographic categories according to the training labels. In the 3rd step the attacker deploys the machine learning model, and then in the 4th step she successfully infers the demographic attributes of the people whose embeddings are stored in the stolen database. In the final 5th step the attacker uses this extracted demographic data to re-identify the people on a social network site.

In this Section, we demonstrate this attack in multiple scenarios, based on the number of people in the database and the accuracy of the demographic data prediction algorithms. In Subsection 4.1 we explain how we generated the data for our experiments, and in Subsection 4.2 we describe the results of our experiment.

4.1 Data generation

To determine the feasibility and threat level of the attack, we ran simulations on the UCI Machine Learning Repository's Adult Dataset [9]. This dataset contains demographic information (including age, sex and race) for more than 30,000 records. These records are not of individual people, but of types of individuals, where the *'fnlwgt'* column describes the number of individuals represented by the given record.

As per our attacker model, our aim with the simulations was to examine what level of re-identification is theoretically possible in a database containing people's

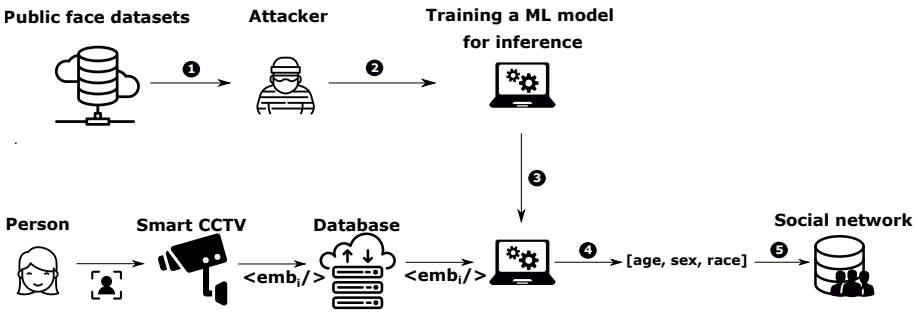


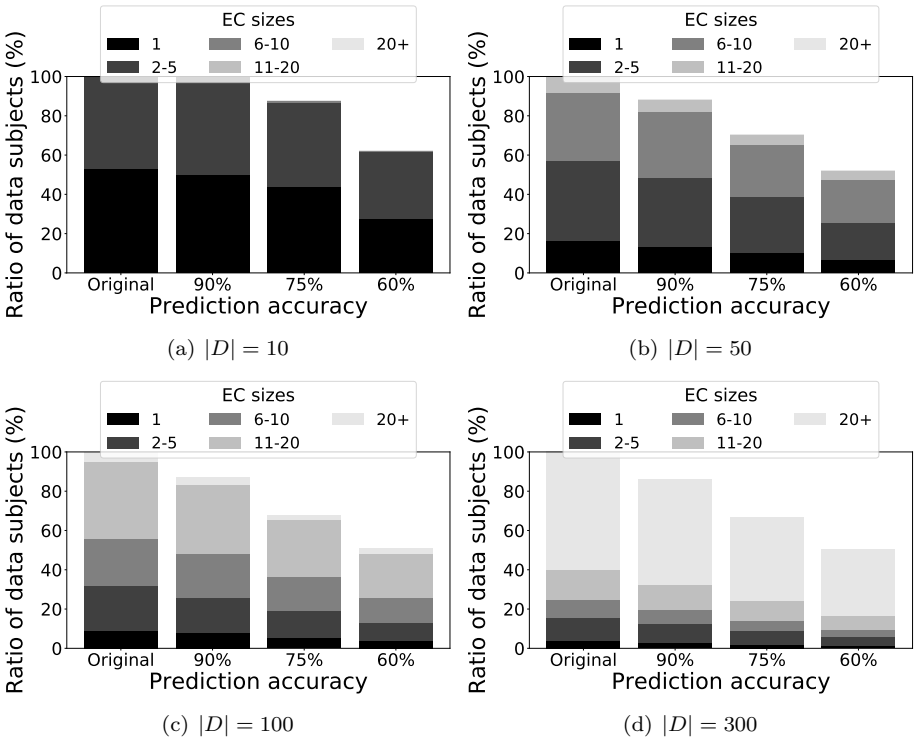
Figure 3: The considered attack when a malicious third party reconstructs demographic data from embeddings and re-identifies the embedding by looking up potential data subjects on social networking sites.

face embeddings. The database sizes were chosen to be reasonable assumptions for the number of employees of a small or medium sized company. To construct the smaller databases of size 10, 50, 100 and 300 for the simulation, we randomly sampled the required number of entries from [9] using the values in the *'fnlwgt'* column as weights, which indicate the number of people represented by a given entry.

4.2 Evaluation

We ran the experiments by assuming the accuracy for predicting age, race and sex to vary between 60%, 75% and 90% and we assumed a machine learning model that can predict age in 10 year intervals. After creating the smaller databases, some of their rows were left untouched based on the prediction accuracy percentages (60%, 75% and 90%), while the remaining rows' age attributes were randomly permuted to simulate inaccurate predictions. This random permutation was then repeated with the same prediction accuracy percentage for the other two attributes, too (sex and ethnicity). This way we ended up with three derived databases for each smaller database, where all three attributes were simulated to be predicted with either 60%, 75% or 90% accuracy. As the last step, for each predicted database we counted what percentage of data subjects were correctly predicted to fall in an equivalence class of size 1, 2-5, 6-10, 11-20 and 20+ (where the smaller the equivalence class, the higher the risk of re-identification is). We then repeated this procedure 100 times and averaged out the results.

Figures 4(a)–4(d) show our findings. We can observe that there are many records in unique or small equivalence classes both in smaller ($|D| = 10$, $|D| = 50$) and larger ($|D| = 100$, $|D| = 300$) predicted databases, which poses privacy risks. The attacker is the most successful at re-identification in the case of the smallest database of 10 people, with the highest 90% prediction accuracy, when 50.1% of people fall in a unique equivalence class, and all the others fall in an equivalence class of size 2-5. If the accuracy is decreased to 60%, still 27.7% falls in a unique



(a) $|D| = 10$ (b) $|D| = 50$

(c) $|D| = 100$ (d) $|D| = 300$

Figure 4: The ratio of equivalence classes (EC) in the predicted database (D) for various database sizes and prediction accuracies.

equivalence class, and 33.9% falls in an equivalence class of size 2-5 (see Figure 4(a)). Regarding the largest database of 300 people, 3.75% of individuals are in a unique equivalence class, and 11.79% are in an equivalence class of size 2-5. Even in the worst case scenario for the attacker, which is 60% accuracy for a database of 300, the rate of people in unique equivalence classes does not fall below 1.38%, nor does the rate of people in an equivalence class of size 2-5 fall below 4.64% (see Figure 4(d)). Also, it is worth noting that while the percentage of people re-identified may be lower in the case of large databases, the expected number of people re-identified may still be higher in these cases. So while an increase in database size and a decrease in prediction accuracy results in a decrease in re-identification probability, the risks are not diminished drastically.

In summary, as expected, the smaller the database size, the higher the re-identification risk is, because smaller sized databases have a higher chance of being reconstructed in such a way that people are correctly mapped to an equivalence class of size 1 or 2-5. Indeed, the higher the prediction accuracy, the higher the re-identification risk is, because the higher percentage of people are predicted to be in the correct equivalence class. As a result, due to the privacy risk presented,

the actual achievable prediction accuracy must be examined, which is detailed in Subsection 5.3.

5 Comparison of the state-of-the-art face recognition libraries

5.1 Data generation

We compare three of the most popular open access FR libraries (OpenCV, Dlib and InsightFace) from a face detection, face recognition and face embedding inference point of view, i.e. how accurately can a machine learning model learn to predict demographic data from the embeddings generated by each library. First, we had to generate a dataset of face images. While there are many publicly available face image datasets, for our purposes we needed a dataset that contained photos of a wide diversity of people: people from both sexes, from four different races and from multiple age groups.

To generate our own dataset, we used the publicly available UTKFace dataset [35], which is a large-scale face dataset that met our requirements, because it contains over 20,000 face images with annotations of age, sex and race. Moreover, the images are labeled with file names formatted like *[age]_[gender]_[race]_[date&time]*, where *age* is an integer from 0 to 116, *sex* is 0 for males or 1 for females, and *race* is 0 for whites, 1 for blacks, 2 for asians, 3 for indians or 4 for other races.

While this dataset was a great starting point for our research, it was not perfect, because we needed a more balanced dataset. As a result, we only used 12192 photos from UTKFace, since there were only 1524 photos per each of the eight race-sex pairs that we worked with (males and females paired with whites, blacks, asians and indians). Of course, some classes (e.g.: white males) had more than 1524 photos, but due to our need for a balanced dataset, we had to choose the number of photos per class based on the least represented class. Even though this subset of UTKFace was balanced regarding sex and race, it still was not balanced regarding age. For example people aged between 20 and 40 were overrepresented, while people aged over 50 were underrepresented, etc. (see Figure 5 for more details regarding the age distribution of our dataset).

5.2 Face detection and recognition

To compare the three libraries, we ran their face recognition algorithms on our dataset. We then examined how many faces each library found out of the 12192, along with the number of false negatives (where a library mistakenly did not find a face in an image) and false positives (where a library mistakenly found multiple faces instead of just one). To gain a better understanding of the accuracy of each library on different races and sexes, we also examined the races and sexes where these false negatives or false positives occurred. Table 1 shows our findings.

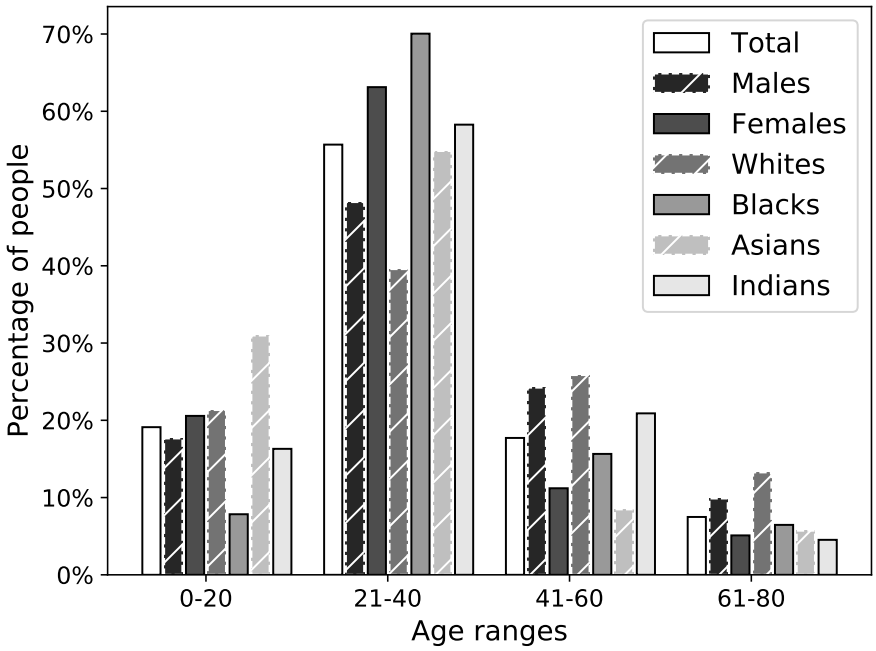


Figure 5: The age distribution of our dataset regarding both sexes and all four examined races

In conclusion, OpenCV and InsightFace performed mostly the same on both sexes and all four races examined, as both libraries had a negligible number of false negatives and false positives. The only difference is the runtime, where OpenCV was about 6 times faster. While Dlib produced zero false positives, it produced a significant false negative rate of 4.2%, which means it is a bit less reliable at detecting people, especially, but not exclusively, black males and females who had a false negative rate of 7.2% and 4.6%. Also, Dlib had the longest runtime, with 4110 seconds, it was about 12 times slower than OpenCV, and 2 times slower than InsightFace.

It is important to note that our tests were conducted on a dataset of cropped face images as opposed to regular face datasets (e.g. "faces in the wild" [18]). So while these results might indicate that there is almost no difference in the false positive rate of the libraries, but due to the nature of our dataset even a very small false positive rate is significant (e.g. in real world conditions OpenCV produces far more false alarms than Dlib [20]). Therefore it is future work to run these experiments on a dataset of non-cropped images "in the wild", too.

In summary, the choice of the right algorithm depends on the use case where facial recognition is applied. When having no false alerts is a significant issue, Dlib is the right choice. When having no false negatives is important, it is better

Table 1: The face detection performance of OpenCV, Dlib and InsightFace on different sexes and races

Lib	**Faces detected**	**Sex**	**Race**	**False pos.**	**Total false pos.**	**False neg.**	**Total false neg.**	**Run time [s]**
Open CV	12183	Male	White	2	3	1	4	322
			Black	1		3		
			Asian	0		0		
			Indian	0		0		
		Female	White	0	2	0	0	
			Black	0		0		
			Asian	1		0		
			Indian	1		0		
Dlib	11676	Male	White	0	0	48	282	4110
			Black	0		110		
			Asian	0		70		
			Indian	0		54		
		Female	White	0	0	51	234	
			Black	0		70		
			Asian	0		62		
			Indian	0		51		
Insight Face	12185	Male	White	0	2	1	4	1858
			Black	1		3		
			Asian	1		0		
			Indian	0		0		
		Female	White	1	1	0	0	
			Black	0		0		
			Asian	0		0		
			Indian	0		0		

to choose another library to avoid situations where some of the consumers could be negatively impacted, such as the previously mentioned incident of Hewlett-Packard's motion-tracking webcams not working on black people [27]. In other cases we may choose between the libraries by considering runtime or the accuracy of additional features. For instance, InsightFace does a great job in detecting facial landmark points, especial when the face is visible from the side profile [6].

5.3 Demographic attribute inference from embeddings

Lastly, we compared each library in terms of how accurately a machine learning model can predict demographic data (sex, race and age) from the face embeddings they produce. The training data was generated by running each library's FR

algorithm on our dataset and collecting the face embedding vectors with their corresponding class labels into *pandas* [23] dataframes, where the labels were deduced from the image file names. Since not all faces were detected in all images by all libraries and since we wanted to train our models on balanced datasets, we had to discard some images in order to always use only as many images per each class as the least represented class permitted (i.e. the class with the lowest number of faces detected).

In total, we built three predictive models per each library, one for sex classification, one for race classification and one for age classification. We used Scikit-Learn's [24] *train_test_split* function to split our dataframes into a train and a test set, and then used Scikit-Learn's *RandomForestClassifier* module to train three random forest classifiers to predict the demographic attributes from the face embeddings. The reason we used random forests was that we wanted to show that even easy to use "off the shelf" ML models can work that do not require deep expertise in ML from an attacker. Random forests satisfy the latter criteria by having a small number of hyperparameters to tune. In the case of age prediction, expecting exact accuracy is not realistic (as it is also difficult for a human to guess the age that precisely), so instead we applied the predictions into ranges between 1-20, 21-40, 41-60 and 61-80 years. Figures 6, 7, 8 show our findings. To evaluate our results, we calculated the prediction F1 score, which is a descriptive metric that takes into consideration the true positive (TP), false positive (FP) and false negative (FN) rates as well: $F1 = \frac{TP}{(TP+\frac{1}{2}\cdot(FP+FN))}$.

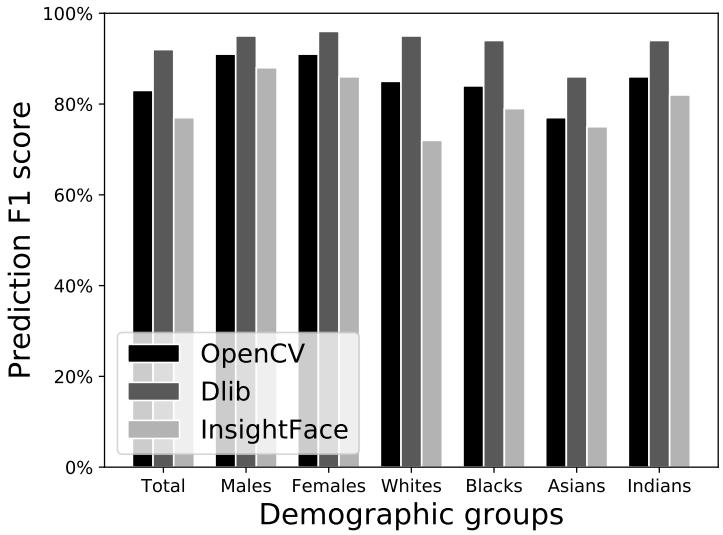


Figure 6: Prediction accuracies for different demographic groups using face embeddings generated by OpenCV, Dlib and InsightFace: Sex prediction

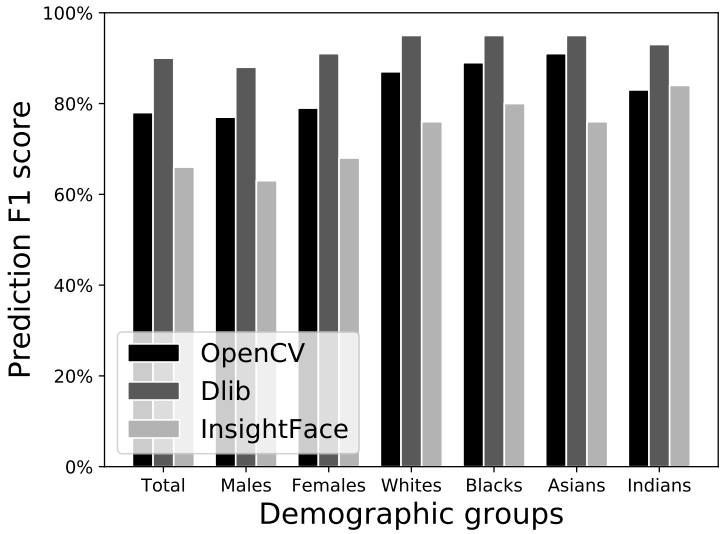


Figure 7: Prediction accuracies for different demographic groups using face embeddings generated by OpenCV, Dlib and InsightFace: Race prediction

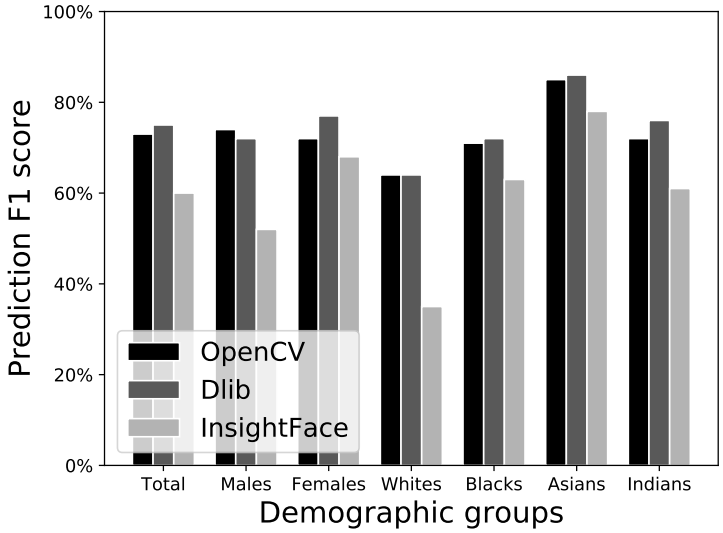


Figure 8: Prediction accuracies for different demographic groups using face embeddings generated by OpenCV, Dlib and InsightFace: Age prediction

In conclusion, our random forest models performed the best on the embeddings generated by Dlib, where the sex classifier achieved over 92%, the race classifier over 89%, and the age classifier over 75% prediction F1 score. The second best performance was achieved when the models were trained and tested on the embeddings generated by OpenCV, where the sex classifier achieved over 83%, the race classifier 78%, and the age classifier over 73% prediction F1 score. The random forest models performed the worst when trained and tested on the embeddings of InsightFace, in which case the sex classifier achieved only over 77%, the race classifier only over 66%, and the age classifier only 60% prediction F1 score.

To test for potential biases, we examined the prediction performance not only for the total population of our dataset, but also on the following smaller demographic groups: males, females, whites, blacks, asian, indians. The performance of the classifiers on these demographic subgroups were mostly uniform, with only a few outliers. While some of the reported differences are very slim, even these could have notable privacy implications as discussed later.

In the case of OpenCV embeddings, the sex classifier performed considerably worse in case of asians than any other race. The race classifier, however, performed the best for asians, and notably worse for indians. The age classifier's performance was significantly worse for white people, but significantly better for asian people. The race prediction performed slightly better for females, whereas the age prediction slightly better for males.

In the case of Dlib embeddings, the sex classifier also achieved a noticeably worse score on asians compared to all other demographic groups. While there was only a very slight difference, but the race classifier achieved the lowest score on the indian population. The age classifier performed the worst on white people, while it performed by far the best on asian people. Regarding sexes, both the race and age predictor performed notably better for females.

In case of the embeddings of InsightFace, the results were a bit different. The sex classifier performed worse than average on whites, and better than average on indians. The race classifier also achieved better than average score for indians, and the lowest score on asians and whites. In the case of age prediction, the most extreme outlier was the much lower score for white people, while the score of asians was also significantly higher than average. In this case, the sex predictor performed slightly better for males, however the race and age prediction was significantly better for females.

Based on these results it seems that there could be noteworthy differences in predicting demographic attributes for different sexes and races. While the impact of these differences may not be significant in all applications (e.g.: targeted advertising in retail), in other scenarios they could have a profound effect on people's lives (e.g.: mass surveillance, law enforcement profiling). Another important aspect to consider is how many people will be affected by the technology in each use case. For example applications in the public sector (e.g.: surveillance by governments) will impact far more people than typical use cases in the private sector (e.g. employee tracking), and in those cases even seemingly small differences of 0.5-1% can affect thousands or tens of thousands of people, which emphasizes the importance of treating facial

recognition technology with great caution.

6 Conclusion and future work

In this paper we have reviewed the main principles behind facial recognition algorithms, introduced three popular Python libraries, and presented the potential discriminational and privacy risks in relation to the processing of face embeddings. We have discussed why face embeddings must be considered sensitive biometric data and we proposed a novel data protection evaluation framework for facial recognition, which could be a general starting point for conducting the DPIA required by the GDPR. We have also looked at various attacker models that could pose a threat to data subjects' privacy via inference based re-identification.

In particular, we analyzed the risks of re-identification by reverse-engineering demographic data (age, sexuality, race) from embeddings stored in a database. We found that the smaller the database and the higher the accuracy of prediction, the higher the re-identification risks are. In the case of a 10 person database and 90% accuracy, 50.1% of people are likely to be precisely re-identified, while this number decreased to 27.7% at 60% prediction accuracy. The risks are also not negligible even for larger databases, because for a database of 300 we showed that at 90% accuracy 3.75% of people are in a unique equivalence class, and 11.79% are in an equivalence class of size 2 to 5 and are likely to be de-anonymized. It must be noted that while the re-identification percentages decrease for larger databases, the absolute number of successful re-identification cases increase.

Afterwards, we compared the performance of three face recognition Python libraries (OpenCV, Dlib, InsightFace) on a custom face image dataset that we have generated. Our findings indicate that while all three libraries produce a negligible number of false positives, Dlib produces far more false negatives than the other two, especially for black people. Regarding run time, OpenCV is about 12 times faster and InsightFace is about 2 times faster than Dlib.

Finally, we extracted face embeddings from our custom dataset using all three libraries to then train random forest classifiers to predict sex, race and age from each library's embeddings. We then compared the prediction accuracies of our random forest models on the total dataset and also on demographic subgroups. Our findings indicate that those models perform the best that were trained and tested on the embeddings of Dlib, followed by the embeddings of OpenCV and finally InsightFace.

Based on our results, there can be differences in prediction accuracies between different sexes and races, and the impact of these biases always has to be evaluated in each application scenario (e.g. law enforcement profiling vs. retail profiling).

For future work, our aim is to gain better understanding and greater explainability of the inner workings of our models in order to discover why misclassifications happen and how demographic data is encoded in embeddings. Also, our focus is to design a procedure that could prevent demographic data leakage from the stored embeddings.

While one obvious approach could be to encrypt the embeddings before storage, they would necessarily have to be decrypted for the calculation of Euclidean or Manhattan distances, so it wouldn't permanently solve the leakage problem. Another plausible solution would be to use homomorphic encryption, which would allow operations to be performed on the embeddings in encrypted form, but due to its computational complexity and slow performance its usage might not be feasible in real-time applications.

Therefore, our research aims find a solution (e.g. adversarial search techniques) to modify the embeddings in such a way to notably lower the prediction accuracies by machine learning models for all demographic and sensitive attributes, without compromising the usability of the face embeddings (i.e.: without significantly changing their relative Euclidean distances). Our hope is that achieving this will allow a much more privacy friendly way to utilize face recognition and process face embeddings.

Acknowledgments

The authors would also like to thank Kenéz Csiktusnádi-Kiss for his work and support in this research.

Icons made by Pixel perfect, catkuro, Eucalyp, fjstudio, Freepik, Pause08, surang, xnimrodx, bimbimkha from `www.flaticon.com`.

References

[1] Bradski, G. The OpenCV Library. *Dr. Dobb's Journal of Software Tools*, 25:120–125, 2000.

[2] Castelluccia, Claude and Le Métayer Inria, Daniel. Impact analysis of facial recognition. Working paper or preprint, URL: `https://hal.inria.fr/hal-02480647`, February 2020.

[3] Chen, Sheng, Liu, Yang, Gao, Xiang, and Han, Zhen. MobileFaceNets: Efficient CNNs for accurate real-time face verification on mobile devices. *ArXiv*, abs/1804.07573, April 2018.

[4] Dalal, N. and Triggs, B. Histograms of oriented gradients for human detection. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 1, pages 886–893, 2005. DOI: `10.1109/CVPR.2005.177`.

[5] Dantcheva, Antitza, Elia, Petros, and Ross, Arun. What else does your biometric data reveal? A survey on soft biometrics. *IEEE Transactions on Information Forensics and Security*, 11, 2015. DOI: `10.1109/TIFS.2015.2480381`.

[6] Deng, Jiankang. Video face recognition demo of ArcFace, 2018. URL: `https://www.youtube.com/watch?v=y-D1tReryGA&ab_channel=JiankangDeng`.

[7] Deng, Jiankang, Guo, Jia, Xue, Niannan, and Zafeiriou, Stefanos. ArcFace: Additive angular margin loss for deep face recognition. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4685–4694, 2019. DOI: 10.1109/CVPR.2019.00482.

[8] Deng, Jiankang, Guo, Jia, Yuxiang, Zhou, Yu, Jinke, Kotsia, Irene, and Zafeiriou, Stefanos. RetinaFace: Single-stage dense face localisation in the wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5203–5212, June 2020.

[9] Dua, Dheeru and Graff, Casey. UCI machine learning repository, 2019. URL: http://archive.ics.uci.edu/ml.

[10] European Parliament and of the Council. Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 april 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing directive 95/46/EC (General Data Protection Regulation), 2016. URL: https://eur-lex.europa.eu/eli/reg/2016/679/oj.

[11] Fábián, István and Gulyás, Gábor György. On the privacy risks of large-scale processing of face imprints. In *The 12th Conference of PhD Students in Computer Science*, 2020. https://www.inf.u-szeged.hu/~cscs/proceedings.php.

[12] Fábián, István and Gulyás, Gábor. De-anonymizing facial recognition embeddings. *Infocommunications Journal*, 12:50–56, 2020. DOI: 10.36244/ICJ.2020.2.7.

[13] GDPR. Article 29 Data Protection Working Party, opinion 05/2014 on anonymisation techniques, 2014. URL: https://ec.europa.eu/justice/article-29/documentation/opinion-recommendation/files/2014/wp216_en.pdf.

[14] Grother, P., Ngan, M., Hanaoka, K., and National Institute of Standards and Technology (U.S.). *Face Recognition Vendor Test (FVRT): Part 3, Demographic Effects*. NIST interagency report. National Institute of Standards and Technology, 2019.

[15] He, Kaiming, Zhang, Xiangyu, Ren, Shaoqing, and Sun, Jian. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. DOI: 10.1109/cvpr.2016.90.

[16] Hill, Kashmir. Another arrest, and jail time, due to a bad facial recognition match. *The New York Times*, 2020. URL: https://www.nytimes.com/2020/12/29/technology/facial-recognition-misidentify-jail.html.

[17] Huang, Gao, Liu, Zhuang, Van Der Maaten, Laurens, and Weinberger, Kilian Q. Densely connected convolutional networks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2261–2269, 2017. DOI: 10.1109/CVPR.2017.243.

[18] Huang, Gary B., Ramesh, Manu, Berg, Tamara, and Learned-Miller, Erik. Labeled faces in the wild: A database for studying face recognition in unconstrained environments. Technical Report 07-49, University of Massachusetts, Amherst, October 2007.

[19] Keegan, Matthew. Big brother is watching: Chinese city with 2.6m cameras is world's most heavily surveilled. *The Guardian*, 2019. URL: https://www.theguardian.com/cities/2019/dec/02/big-brother-is-watching-chinese-city-with-26m-cameras-is-worlds-most-heavily-surveilled.

[20] King, Davis. dlib vs OpenCV face detection, 2014. URL: https://www.youtube.com/watch?v=LsK0hzcEyHI.

[21] King, Davis E. Dlib-ml: A machine learning toolkit. *Journal of Machine Learning Research*, 10(60):1755–1758, December 2009.

[22] Mai, Guangcan, Cao, Kai, Yuen, Pong C., and Jain, Anil K. On the reconstruction of face images from deep face templates. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(5):1188–1202, May 2019. DOI: 10.1109/tpami.2018.2827389.

[23] McKinney, Wes. Data structures for statistical computing in python. *Proceedings of the 9th Python in Science Conference*, 445:51–56, 2010. DOI: 10.25080/majora-92bf1922-00a.

[24] Pedregosa, Fabian, Varoquaux, Gaël, Gramfort, Alexandre, Michel, Vincent, Thirion, Bertrand, Grisel, Olivier, Blondel, Mathieu, Prettenhofer, Peter, Weiss, Ron, Dubourg, Vincent, et al. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12(85):2825–2830, 2011.

[25] Ramirez Cerna, Lourdes, Camara-Chavez, Guillermo, and Menotti Gomes, David. Face detection: Histogram of oriented gradients and bag of feature method. In *Proceedings of the 2013 International Conference on Image Processing, Computer Vision, and Pattern Recognition (IPCV'13)*, 2013. https://worldcomp-proceedings.com/proc/p2013/IPC.html.

[26] Schroff, Florian, Kalenichenko, Dmitry, and Philbin, James. FaceNet: A unified embedding for face recognition and clustering. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, Jun 2015. DOI: 10.1109/cvpr.2015.7298682.

[27] Simon, Mallory. HP looking into claim webcams can't see black people. *CNN*, 2009. URL: https://edition.cnn.com/2009/TECH/12/22/hp.webcams/index.html.

[28] Sweeney, Latanya. Simple demographics often identify people uniquely. *Carnegie Mellon University, Data Privacy*, 2000. DOI: https://doi.org/10.1184/R1/6625769.v1.

[29] Szegedy, Christian, Ioffe, Sergey, Vanhoucke, Vincent, and Alemi, Alexander A. Inception-v4, Inception-ResNet and the impact of residual connections on learning. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, AAAI'17, pages 4278–4284. AAAI Press, 2017.

[30] Thorat, S. B., Nayak, S. K., and Dandale, Jyoti P. Facial recognition technology: An analysis with scope in India. *ArXiv*, abs/1005.4263, 2010.

[31] Turk, M.A. and Pentland, A.P. Face recognition using eigenfaces. In *Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 586–591, 1991. DOI: 10.1109/CVPR.1991.139758.

[32] Viola, P. and Jones, M. 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)*, volume 1, 2001. DOI: 10.1109/CVPR.2001.990517.

[33] Wu, Haoran, Xu, Zhiyong, Zhang, Jianlin, Yan, Wei, and Ma, Xiao. Face recognition based on convolution siamese networks. In *10th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI)*, pages 1–5, 2017. DOI: 10.1109/CISP-BMEI.2017.8302003.

[34] Zhang, Maggie. Google Photos tags two African-Americans as gorillas through facial recognition software. *Forbes*, 2015. URL: https://www.forbes.com/sites/mzhang/2015/07/01/google-photos-tags-two-african-americans-as-gorillas-through-facial-recognition-software/.

[35] Zhang Zhifei, Song, Yang and Qi, Hairong. Age progression/regression by conditional adversarial autoencoder. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2017. DOI: 10.1109/cvpr.2017.463.