Efficient Aggregation of Face Embeddings for Decentralized Face Recognition Deployments

Philipp Hofer[1], Michael Roland[1], Philipp Schwarz[2]and René Mayrhofer[1]
[1]*Johannes Kepler University Linz, Institute of Networks and Security, Austria*
[2]*Johannes Kepler University Linz, LIT Secure and Correct Systems Lab, Austria*

Keywords: Biometric Authentication, Face Embedding, Face Recognition, Aggregation, Decentralization.

Abstract: Ubiquitous authentication systems with a focus on privacy favor decentralized approaches as they reduce potential attack vectors, both on a technical and organizational level. The gold standard is to let the user be in control of where their own data is stored, which consequently leads to a high variety of devices used what in turn often incurs additional network overhead. Therefore, when using face recognition, an *efficient* way to compare faces is important in practical deployments. This paper proposes an efficient way to aggregate embeddings used for face recognition based on an extensive analysis on different datasets and the use of different aggregation strategies. As part of this analysis, a new dataset has been collected, which is available for research purposes. Our proposed method supports the construction of massively scalable, decentralized face recognition systems with a focus on both privacy and long-term usability.

1 INTRODUCTION

Applications processing personal data do not have to disclose their existence, thus it is not possible to know about every data-processing system. However, the public knows about at least some of these systems, as they have an immediate effect on individuals: There are many databases featuring an extensive amount of highly personal data already in production for many years, such as the Indian Aadhaar system. These biometric recognition deployments are prime examples of city- or nation-scale ubiquitous systems that already bridge the digital and physical worlds through their use of (biometric and other) sensor data for deriving decisions about which physical world interactions people are authorized for. Especially due to the trend of increasing quantity and quality of such systems, it is especially critical to think about systems that contain highly sensitive personal data — even more so, as these systems may depend on public acceptance.

In general, data can be stored in a central location or in a decentralized manner, i.e. spread across multiple locations. Unfortunately for individuals' privacy, the backbone for most current systems is a central database with highly sensitive personal data, such as biometrics. This makes such systems especially vulnerable to multiple attack vectors: trusting the provider and its security measures. Unfortunately, even under highest security constraints, data breaches are permanently happening, even (or especially) with the largest entities.

In order to reduce this massive attack surface of having everything stored in a single place, we instead propose to design decentralized systems. The individual should ideally be allowed to choose who is managing their (personal) data (or store them on their own devices) and have the chance to move from one service provider to another.

However, these kinds of distributed systems suffer from drawbacks. The integrity of each device that manages an individual must be verified to prevent identity theft by malicious devices. Stressing (hardware) requirements might favor fewer, big players who have the necessary resources, and therefore systems with fewer requirements will support a larger provider distribution. Since biometrics are one of the most privacy-sensitive data points and efficiency is crucial for the successful deployment of large-scale systems, this paper focuses on the computational complexity required for biometric authentication systems.

Modern face recognition typically start by extracting embeddings that represent feature vectors of faces through the use of machine learning models. Subsequently, they calculate the similarity to all of these

Hofer, P., Roland, M., Schwarz, P. and Mayrhofer, R.
Efficient Aggregation of Face Embeddings for Decentralized Face Recognition Deployments.
DOI: 10.5220/0011599300003405
In *Proceedings of the 9th International Conference on Information Systems Security and Privacy (ICISSP 2023)*, pages 279-286
ISBN: 978-989-758-624-8; ISSN: 2184-4356

images by again deriving embeddings and comparing them to previously stored templates and each other. This is inefficient on two different levels: First, requiring multiple (on a holistic system point of view, redundant) similarity calculations would hurt both provider diversification and hinder small providers in serving larger quantities of users because of increased hardware requirements. Ideally, one could combine different aspects of these multiple embeddings extracted from face images with as little data as possible. Since research on still image face recognition is quite extensive, and an embedded camera sensor device can often derive embeddings of the currently visible person on-line, creating a new, aggregated embedding based on all images available of an individual would not change the backbone of state-of-the-art face recognition pipelines. Secondly, having a single (aggregated) embedding, thus not depending on multiple similarity computations minimizes network traffic, which is especially significant for decentralized, embedded systems.

In this paper, our focus is on evaluating different methods of aggregating face embeddings (Section 3) from an efficiency and accuracy point of view. Furthermore, we test the limit of sufficient image quantity and analyze whether there is a clear point where adding additional images does not significantly increase face recognition accuracy. Last but not least, in order to verify if using multiple images in different settings boosts accuracy significantly, we propose a new in-the-wild dataset, where subjects take around 50 images of themselves in a single setting, which only takes around 3 seconds to achieve practical usability. Additional images in radically different settings are used as approximation of the true embedding to verify the performance improvements.

2 MULTI-IMAGE FACE RECOGNITION

In order to evaluate and compare different face recognition methods, they are tested against public datasets. Many of these face recognition datasets typically are high quality (to facilitate training face recognition models) and high quantity (to reduce the bias).

Most datasets define a fixed set of pairs of images to allow for objective evaluation of face recognition methods. With this strategy, a single image is used as template in state-of-the-art face recognition pipelines. This template is then compared with positive (same person) and negative (different person) matches. This approach tests one important metric of face recognition: How well it performs on still images. Compared to more complex scenarios, only testing on still images is efficient at runtime, which decreases computation time to evaluate the accuracy on a dataset. However, there are different aspects this method does not test, such as how to handle multiple images or even video streams of a person.

In reality, these ignored aspects are essential, as live-images from cameras do not produce high-quality images similar to the images from many available and commonly used face recognition datasets. Instead, the person-camera angle is far from optimal, the person is not directly in front of the camera and thus the face is quite small. Furthermore, the face can be occluded, e.g. with a scarf, sunglasses, or hair. In these real-world settings, face recognition pipelines have a harder time recognizing people than with public datasets, although new datasets try to represent these challenges. Nevertheless, there is a potential benefit of real-world scenarios: There are many images of a single person available, as the person is presumably visible for (at least) many seconds and thus a camera is able to capture significantly more than one image.

One way of bridging the gap of having multiple images of the same person and being of lower quality is to merge the embeddings obtained from multiple images into a single embedding. More accurate templates, by definition, lead to accuracy improvements of face recognition. The idea behind using multiple images is that it is not possible to capture a perfect representation of a face in a single picture due to various reasons (occlusion, lighting, accessories, ...).

While a single image cannot account for all these different settings, multiple images can capture different face areas and settings. Therefore, using multiple images provides more information about the individual's face, and we therefore expect an increased accuracy. As introduced in Section 1, due to hardware and network constraints, comparing the current live-image with multiple embeddings of the same person is not favorable in some situations. For an efficient face recognition pipeline, it would be best to only have a single embedding which is used as template for a person. This would allow the system to make use of the vast literature on single face image recognition.

In contrast to this single-embedding approach, in recent years other work is published in the domain of video face recognition (Rao et al., 2017; Rivero-Hernández et al., 2021; Zheng et al., 2020; Liu et al., 2019; Gong et al., 2019). Most of these papers propose an additional neural network to perform the weighting of different embeddings (Liu et al., 2019; Rivero-Hernández et al., 2021; Yang et al., 2017; Gong et al., 2019). Especially on embedded devices,

these additional networks have a significant runtime impact, as they need to perform an additional inference step. In order to be runtime-efficient even on embedded hardware, this paper focuses on creating a single embedding.

In state-of-the-art face recognition tools, embeddings are high-dimensional vectors. If multiple embeddings should be aggregated to a single one, this opens up the questions:

RQ1. How do we (numerically) *best* aggregate the embeddings, and is this aggregation actually increasing face recognition performance? How can we define *best*?

RQ2. After knowing how to aggregate embeddings, how many images are necessary and useful? Is there a point from which adding additional embeddings do not significantly increase accuracy?

Depending on the application, there may or may not be a lot of data available for each person. Therefore, in many situations (e.g. enrollment of a user) it could make the process significantly easier if the data can be recorded in a single session and therefore featuring only one setting. This leads to the question:

RQ3. Is it beneficial to use different settings? Is it worth creating images with and without (typical) accessories, such as face masks, glasses, and scarfs?

Similarly, it might be unrealistic to expect many images in various settings from a new user. Having to verify that these different settings really belong to the same person makes it even more complicated. It is easy and practical to capture a couple of images in one place.

RQ4. Is it enough if we use only images while we rotate our heads for the aggregated embedding, similar to the process of how some smartphone enroll users' faces? Is the accuracy increased if we include totally different settings in the aggregated embedding?

3 EMBEDDING AGGREGATION

This paper evaluates different aggregation strategies and proposes efficient ways of aggregating embeddings in order to create a single, efficient template-embedding containing as much information as possible. If multiple images of a person are used, the position of the person of interest has to be extracted in each frame. These positions could be either fed to a neural network which expects multiple images (or a video) as input (*video-based face recognition*) or the embedding could be extracted for each frame and then aggregated (*imageset-based face recognition*).

Video face recognition networks have to perform all necessary steps in a single network. Extracting the embeddings frame-by-frame and only then aggregating these embeddings to a single template allows for a much more modular pipeline. This goes hand-in-hand with traditional face recognition pipeline approaches, which can be separated into face detection, face tracking, and face recognition, and therefore also allow for being able to individually optimize each part. Additionally, systems using this approach can use its vast literature, as the field of (still) image face recognition is much more advanced than video face recognition. Therefore, in this paper, we focus on the modular approach of extracting embeddings from every image and then aggregating them.

Since we want to aggregate multiple embeddings extracted from single frames, we need an *aggregation strategy*. Literature typically calculates the mean of each dimension of the embedding, e.g. as proposed by Deng et al. (Deng et al., 2019):

$$\begin{pmatrix} a_1 \\ \vdots \\ a_n \end{pmatrix}, \begin{pmatrix} b_1 \\ \vdots \\ b_n \end{pmatrix} \rightarrow \begin{pmatrix} \text{mean}(a_1, b_1) \\ \vdots \\ \text{mean}(a_n, b_n) \end{pmatrix}$$

Fig. 3 shows the instance space of two people. The x- and y-axis represent the PCR-reduced form of their embeddings. The triangle represents the average of each dimension of the embedding for each person. There is no analysis on whether it is useful to use the mean of each dimension, or if there are better approaches to aggregate embeddings. There is not even an analysis if calculating the mean of the embeddings improves the accuracy of face recognition pipelines. In order to verify this hypothesis, a baseline is needed to compare the performance of aggregated embeddings to. In this paper, we use pre-trained state-of-the-art face-detection (RetinaFace (Deng et al., 2020)) and -recognition (Arcface (Deng et al., 2019)) models. Arcface receives a single image as input and creates a 512-dimensional vector. Even though we did not explicitly test different architectures, we expect similar results on semantically similar networks. Further work will focus on extending these experiments to multiple face detection and -recognition models.

3.1 Dataset

We chose to evaluate the face recognition models on the CelebA dataset (Liu et al., 2015) because it contains multiple images of thousands of people.

More specifically, the dataset contains 10.177 people. 2.343 people have exactly 30 images.

In order to reduce the chance of having outliers, we remove all images with less than 30 images. For consistency, we also remove people with more than 30 images. Furthermore, we cleaned the dataset by performing face detection with RetinaFace. From the initial 2.343 people with 30 images, there are 20 people which contain an image where face detection could not detect a face—mainly due to too much occlusion. 18 randomly chosen images, where face detection did not work, are shown in Fig. 1. In order to have a dataset as consistent as possible, we removed all images of these 20 people, resulting in a final set of 2.323 people.

The CelebA dataset has been pre-processed such that the main person is in the center of the image, so if multiple faces are detected, we take the most central person and ignore the remaining ones.

In total, there are 30 images of 2.323 people each in our cleaned dataset, resulting in a total number of 69.690 images. In order to objectively evaluate the difference between different aggregation strategies, we reserve 10 random images of each person as potential template images. Since the dataset does not have a specific order, without loss of generality and for reproducibility, we reserve the first 10 images as potential template images.

In the first setting (*baseline*), we (only) use the embedding of the first image.

Calculating the mean of different embeddings is only one possible strategy to aggregate embeddings. For different methods, such as taking the minimum of each dimension, we numerically aggregate each dimension of the embeddings from images 1 – 9, using its respective aggregation strategy. Applications set a threshold for their specific task, under which two faces are recognized as the same person. This decision is based on their safety requirements. For security-critical applications the threshold should be lowered, which results in fewer false-positives (but potentially more false-negatives).

In order to compare the strategies, we take the average distance of the template to each embedding from images 11 – 30 as our metric:

$$err = \frac{\sum_{p \in people} \sum_{emb_{test} \in testEmbs} dist(emb_{test}, p_{template})}{len(people) \times len(testEmbs)}$$

Figure 1: Example images where RetinaFace could not detect a person.

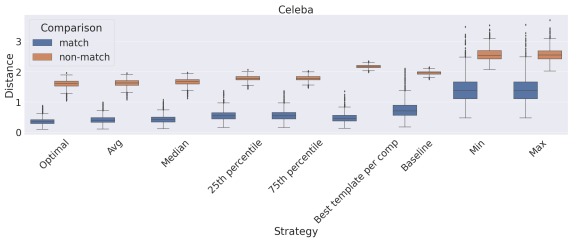


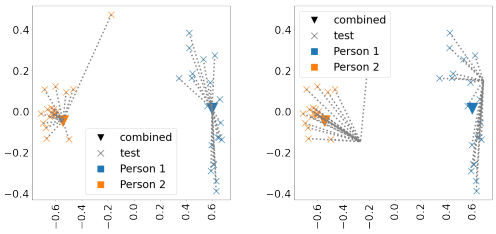
Figure 2: Average distance of template- to test-embeddings in CelebA dataset.

A smaller *error* represents a higher confidence of the network, that the template belongs to the test images. Semantically, the error specifies the average distance of the template to each test image.

The *CelebA* row in Table 1 shows the resulting distance between the template- and test-embeddings. Fig. 2 visually represents the CelebA column. With respect to our 2: Except for the (cheating) *optimal* setting (which we discuss later), the best aggregation strategy is using the mean of every dimension of the embedding. As the distance compared to the baseline is significantly lower in the *average* (and *median*) setting, this clearly shows the effective impact of using multiple (in our case 10) images as templates. Aggregating multiple embeddings using the mean significantly outperforms the baseline. Fig. 3 shows the intuition behind this behavior on the first two people. If more than one image of a person is used, the resulting embedding approximates the optimal embedding more accurately. The optimal embedding of a person with respect to the current test images, is the average of these test embeddings, since this would minimize the respective distance.

So far, it was shown that using the average of 10 images significantly outperforms a single image used as template. Naturally, the question arises if the accuracy still increases if more images are used. Is there a limit above which additional images will not further improve accuracy (2)?

To answer this question, we need a dataset with more images of the same person. For this purpose we used the LFW dataset (Huang et al., 2008) as it



(a) The mean-template. (b) The first image.

Figure 3: Distance of the embeddings to...

Table 1: Average distances of the template to the test embeddings under different aggregation strategies (factor to baseline). For *match*es (top line in each cell), a higher factor is favorable, while for *non-match*es (bottom line), a lower factor is better. The gray rows use information usually not available, and are displayed for comparison reasons only.

	CelebA	PS-Normal	PS-Smile
Baseline	0.7 (1.0x) 1.9 (1.0x)	0.6 (1.0x) 1.8 (1.0x)	0.6 (1.0x) 1.8 (1.0x)
Avg	0.4 (1.8x) 1.6 (1.2x)	0.4 (1.5x) 1.5 (1.2x)	0.4 (1.4x) 1.4 (1.2x)
Median	0.4 (1.8x) 1.6 (1.2x)	0.4 (1.5x) 1.6 (1.2x)	0.4 (1.4x) 1.5 (1.2x)
Min	1.4 (0.5x) 2.5 (0.8x)	2.2 (0.3x) 3.1 (0.6x)	2.1 (0.3x) 3.0 (0.6x)
Max	1.4 (0.5x) 2.5 (0.8x)	2.1 (0.3x) 3.0 (0.6x)	2.1 (0.3x) 3.0 (0.6x)
Optimal	0.3 (2.1x) 1.6 (1.2x)	0.3 (1.6x) 1.5 (1.3x)	0.3 (1.6x) 1.4 (1.3x)
Best template	0.4 (1.6x) 2.1 (0.9x)	0.1 (3.6x) 2.1 (0.9x)	0.2 (2.9x) 2.1 (0.9x)

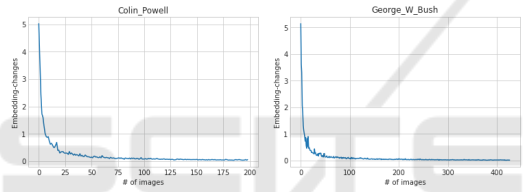


Figure 4: Numeric embedding differences shown for 2 people from the LFW dataset.

contains hundreds of images of the same people. In particular, we use the 5 people in the LFW dataset who have more than 100 images.

For each person, the embedding of the first image serves as the starting point. Next, the embedding of the second image is extracted. The first point of each plot in Fig. 4 represents the sum of the difference between these two embeddings. We then combine all previously used embeddings into our template. Afterwards, we extract the embedding of the next image, calculate its difference to the template, and plot the value. We continue with this approach until we used every available image.

Interestingly, it looks like a (fuzzy) inverse log function. Intuitively, this makes sense as new images contain a lot of new information in the beginning, but after the template consists of many aggregated images, a new image cannot provide as much new information as in the beginning. Furthermore, there seems to be a limit of roughly 50 images, after which the embedding is not changing significantly anymore. Another aspect to point out, is that there are some *upticks* in the graph. After looking at the specific images which cause these effects, they all present a new variation of the face (either a new face-angle or different accessories).

Section 3 used images of the same person in different settings, such as different hairstyles, lighting, and location. For the most part, the dataset consists of frontal images as the person is directly looking into the camera. Some modern smartphones provide the ability to unlock the phone by rotating the phone around the head. This is probably not only used to detect the liveness of the person, but also to increase the amount of information gained from the camera. Is the difference in angle from this type of recording enough to utilize the benefit of combining embeddings discussed so far (2)?

Therefore, we did a similar analysis on a different dataset: Pan Shot Face Database (PSFD) (Findling and Mayrhofer, 2013). This dataset features 30 participants from 9 perspectives. Every perspective contains 5 *look directions* (straight, slightly top left, slightly top right, slightly bottom right, and slightly bottom left) and 4 distinct *facial expressions* (normal, smiling, eyes closed, and mouth slightly opened). This gives us 5,400 images to work with.

For the first test, we used all images with a *normal* face expression as template and evaluated its average distance to all other images. The result is visible in the *PS-Normal* row in Table 1.

People in this dataset seem to be easier recognized compared to the CelebA dataset, which is reflected in a lower average distance (Table 1). For the CelebA dataset, the template which consists of 10 images is performing 1.8 times better than if only a single image is used as template. Interestingly, on our new dataset this improvement is in the same order of magnitude: 1.5 times better.

In order to simulate real-world templates, images are professional portrait photographs (e.g. used as profile images) of the subject. In the second scenario templates are created with images of the smiling person. The outcome of this *PS-Smile* setting, is not significantly different to the original *PS-Normal* setting (c.f. Table 1). Thus, it does not make a significant difference which facial expression the person put on while creating template images.

4 SINGLE SETTING PERFORMANCE

In our experiments so far, we used images of the same people in different settings, as these are the most common images provided by available datasets. In practice, however, it would seem convenient for both

the provider and the individual to only use images taken at the time of physical enrollment. The provider would benefit by ensuring that the individual is not spoofing the system, e.g., by using images from other people which would break security guarantees for all kinds of authentication systems, both with publicly issued credentials such as passports and with accounts enrolled with only a single (e.g., building access control) system. The advantage for the user is better usability, as they do not have to provide any additional data besides their participation in the enrollment procedure. With enrollment interaction limited to a few seconds, we argue that creating a more diverse set of input face images for improving recognition accuracy as proposed in this paper takes less effort than creating a traditional user account with setting a new password.

Unfortunately, there are no publicly available datasets that systematically contain both images of people in the same setting (e.g., only rotating the head, as performed for some mobile phone face authentication implementations) and also images in different settings. In order to test our hypothesis of only using a single setting while additional images of the same person in different settings do not increase accuracy, we created a new dataset, which we called *In-The-Wild Face Angle Dataset*. We will also use this dataset to answer 2. Inspired by Datasheets for Datasets (Gebru et al., 2021), we describe the dataset on our website: digidow.eu/experiments/face-angle-dataset.

In order to test the increased performance if multiple images recorded in a single session are used, we calculated a rolling average of the template images. The first data point for each person is equal to the first embedding. The second data point is the average of the first two embeddings. The last data point is the average of all embeddings of this particular person.

In order to quantify the performance, the 10 images of each person in different settings are not used as template images. Instead, the average distance between the rolling average of the template images and test images is calculated.

The average distance between the first image of every person and their test images is, on average, 0.699. If we not only use the first image, but rather the average of all template images, the average distance drops significantly to 0.291. Fig. 5 shows the average plot of a person.

Interestingly, this opposes the previous findings as the distance grows smaller even after the limit of roughly 50 images. We argue, that this is due to the fact that not the amount of images, but the amount of **semantically different** images are important.

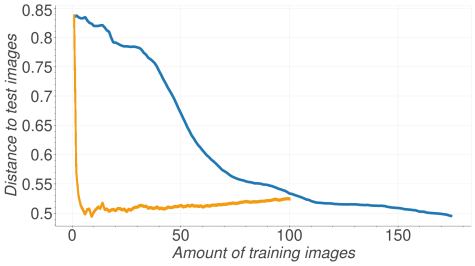


Figure 5: Rolling distance average of the aggregated embedding to the test images. The y-axis shows the average distance to the test images (orange → greedy search; blue → ordered).

To verify this, we ran the same experiment, but used only every n^{th} image as template image. The distance decreases if more images are used (all images: 0.291, $n = 10$: 0.294, $n = 20$: 0.297, $n = 50$: 0.325, single image: 0.699). However, distance improvements are certainly not linear and are leveling off at some point. The improvement from using just a few images is only marginally better than using hundreds of images suggests, that the amount of images only plays a minor role.

If the improvement best seen in Fig. 2 is due to having different angles of the face, we expect a similar improvement if we switch from using dozens to just a few images picturing different angles. Therefore, instead of using the template images in sequence, a greedy search on every iteration should result in the best embedding for each step. On every step, we create the new average embedding for all remaining images of the person, calculate the new distance to the test images and select the one which minimizes this distance. After using just the 3 best images, accuracy already improved significantly and there is little room for improvement (0.315 for the 3 best images vs 0.291 if all images are used). After manually inspecting the top images for each person, in 82 % of the cases the first 3 images are one frontal image, and two profile images from each side. Further work could add convergence criteria to automatically select the best amount of images.

5 RELATED WORK

Chowdhury et al. (Chowdhury et al., 2016) proposed an interesting change: Instead of using the mean-weighting of features, they propose to use the maximum instead. This should reduce the overfit on dominant angles and generalize better (Chowdhury et al., 2016). However, this could not be replicated with this dataset, as the *minimum* and *maximum* settings perform significantly worse than the baseline (Ta-

ble 1). One potential cause for this bad performance is that outliers have too much impact on the final template. Therefore, we created another template by using $\{25,75\}th$ quantile of each dimension of the embedding, which scores significantly better than both the *minimum* and *maximum* setting, but not as well as the *average* aggregation strategy.

Rao et al. (Rao et al., 2017) created a pipeline with a similar goal. Instead of aggregating the embeddings into a single template, they created a neural network which receives raw images as input. As the networks have full access to the whole image (instead of an embedding only), this approach offers the possibility of higher accuracy on the drastic expense of runtime-performance and is thus not really suitable for embedded systems.

Furthermore, in the last years, a lot of effort is spent on deciding how to weigh different dimensions of embeddings (Yang et al., 2017; Rivero-Hernández et al., 2021; Liu et al., 2019). Even though some of these approaches look promising, they are not ideal for embedded systems, as most of them use additional hardware-intense computations. Therefore, this work does not favor any specific image over another.

Balsdon et al. (Balsdon et al., 2018) showed that accuracy of humans doing face identification significantly improves in a "wisdom of crowd" setting compared to individual's performance. This could indicate, that a similar effect is demonstrable if a system combines embeddings not only from a single face recognition neural network, but from multiple different ones. Therefore, further work could use the proposed method of combining embeddings of different neural networks, potentially using the same aggregation strategies as analyzed in the present paper.

6 CONCLUSION

In this work, we evaluated different aggregation strategies, leading to the conclusion that aggregating embeddings by taking the average of each dimension provides the highest improvement in accuracy while remaining compatible to state-of-the-art face recognition pipelines as already widely deployed in the field. We stress that this was one of the design goals of our work, and that our results indicate that such improvements can be directly applied to existing (embedded and distributed) systems with changes to only the enrollment and template computation processes, but not the live recognition pipelines.

Even though some previous work implicitly used this average aggregation strategy, there has been no evaluation about its effectiveness. We base this proposal on an extensive evaluation of different aggregation strategies using both different public datasets and creating a new dataset, which is publicly available for research purposes. After quantitatively analyzing the number of images used to generate templates, we find that it only plays a minor role, while different perspectives — we refer to them as semantically different input — significantly improve the performance of face recognition pipelines. For an efficient, decentralized system, we propose to use just 3 images per template: one frontal image and one from each side. These images may share the same setting, thus if there is a physical enrollment, these images can be taken live. This increases both the correctness of the system itself (as there are fewer options to spoof the system) and the usability of the system (as the user does not have to provide larger sets of images or even video footage).

Future work could focus on automatically choosing the best images based on various convergence criteria, as well as further studying continuous learning approaches that update the aggregated template with new input as user faces change over time (e.g., with age or clothing). We argue that the average aggregation strategy that results in best results in our study would lend itself optimally to dynamic updates; however, the associated security impact and added threat models by allowing templates to be updated outside a controlled enrollment setting will need to be considered carefully for each scenario.

ACKNOWLEDGEMENTS

This work has been carried out within the scope of Digidow, the Christian Doppler Laboratory for Private Digital Authentication in the Physical World and has partially been supported by the LIT Secure and Correct Systems Lab. We gratefully acknowledge financial support by the Austrian Federal Ministry of Labour and Economy, the National Foundation for Research, Technology and Development, the Christian Doppler Research Association, 3 Banken IT GmbH, ekey biometric systems GmbH, Kepler Universitätsklinikum GmbH, NXP Semiconductors Austria GmbH & Co KG, Österreichische Staatsdruckerei GmbH, and the State of Upper Austria.

REFERENCES

Balsdon, T., Summersby, S., Kemp, R. I., and White, D. (2018). Improving face identification with specialist

teams. *Cognitive Research: Principles and Implications*, 3(1):1–13.

Chowdhury, A. R., Lin, T.-Y., Maji, S., and Learned-Miller, E. (2016). One-to-many face recognition with bilinear cnns. In *2016 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1–9. IEEE.

Deng, J., Guo, J., Ververas, E., Kotsia, I., and Zafeiriou, S. (2020). Retinaface: Single-shot multi-level face localisation in the wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5203–5212.

Deng, J., Guo, J., Xue, N., and Zafeiriou, S. (2019). Arcface: Additive angular margin loss for deep face recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4690–4699.

Findling, R. D. and Mayrhofer, R. (2013). Towards pan shot face unlock: Using biometric face information from different perspectives to unlock mobile devices. *International Journal of Pervasive Computing and Communications*.

Gebru, T., Morgenstern, J., Vecchione, B., Vaughan, J. W., Wallach, H., Iii, H. D., and Crawford, K. (2021). Datasheets for datasets. *Communications of the ACM*, 64(12):86–92.

Gong, S., Shi, Y., Kalka, N. D., and Jain, A. K. (2019). Video face recognition: Component-wise feature aggregation network (c-fan). In *2019 International Conference on Biometrics (ICB)*, pages 1–8. IEEE.

Huang, G. B., Mattar, M., Berg, T., and Learned-Miller, E. (2008). Labeled faces in the wild: A database forstudying face recognition in unconstrained environments. In *Workshop on faces in'Real-Life'Images: detection, alignment, and recognition*.

Liu, Z., Hu, H., Bai, J., Li, S., and Lian, S. (2019). Feature aggregation network for video face recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*.

Liu, Z., Luo, P., Wang, X., and Tang, X. (2015). Deep learning face attributes in the wild. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3730–3738.

Rao, Y., Lu, J., and Zhou, J. (2017). Attention-aware deep reinforcement learning for video face recognition. In *Proceedings of the IEEE international conference on computer vision*, pages 3931–3940.

Rivero-Hernández, J., Morales-González, A., Denis, L. G., and Méndez-Vázquez, H. (2021). Ordered weighted aggregation networks for video face recognition. *Pattern Recognition Letters*, 146:237–243.

Yang, J., Ren, P., Zhang, D., Chen, D., Wen, F., Li, H., and Hua, G. (2017). Neural aggregation network for video face recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4362–4371.

Zheng, J., Ranjan, R., Chen, C.-H., Chen, J.-C., Castillo, C. D., and Chellappa, R. (2020). An automatic system for unconstrained video-based face recognition. *IEEE Transactions on Biometrics, Behavior, and Identity Science*, 2(3):194–209.