# Improving deep-learning-based face recognition to increase robustness against morphing attacks

Una M. Kelly, Luuk Spreeuwers and Raymond Veldhuis

Data Management and Biometrics Group, University of Twente, The Netherlands

***Abstract***

*State-of-the-art face recognition systems (FRS) are vulnerable to morphing attacks, in which two photos of different people are merged in such a way that the resulting photo resembles both people. Such a photo could be used to apply for a passport, allowing both people to travel with the same identity document. Research has so far focussed on developing morphing detection methods. We suggest that it might instead be worthwhile to make face recognition systems themselves more robust to morphing attacks. We show that deep-learning-based face recognition can be improved simply by treating morphed images just like real images during training but also that, for significant improvements, more work is needed. Furthermore, we test the performance of our FRS on morphs of a type not seen during training. This addresses the problem of overfitting to the type of morphs used during training, which is often overlooked in current research.*

***Keywords***

*Biometrics, Morphing Attack Detection, Face Recognition, Vulnerability of Biometric Systems*

## 1. Introduction

A Face Recognition System (FRS) performs identity verification by comparing two photos and deciding whether or not the identities match. It was first shown in [1] that existing Face Recognition Systems (FRS) were vulnerable to *morphing* attacks. A morph is an image that contains facial features of two different people. In a border-crossing scenario a criminal (C) could enlist the help of an accomplice to create a morphed photo. The accomplice (A) could then use this photo to apply for a passport, which the criminal in turn could use to cross borders undetected. The most-used method to create morphs is to mark certain facial features, called landmarks, warp both images to a common geometry and then blend the pixel values. For an overview of this morphing process see Fig. 1. It has been shown that both FRSs and humans will often accept a morph made with this method as a match with both contributing identities [2–4].

Recently, a platform was launched with which the performance of different morphing detection algorithms can be benchmarked [5]. This benchmark and other research indicates that existing algorithms do not perform well when tested across different datasets [6, 7]. Since researchers have so far had to create their own training datasets, their detection methods may have been overfitted to specific characteristics of their training set. Furthermore, some detection methods require large datasets for training, which means that a large number of morphs need to be made. Since this is usually done automatically such morphs will probably be of lower quality than hand-crafted morphs. A detection method created by training with such data can detect low-quality but not high-quality morphs.

David C. Wyld et al. (Eds): SPPR, NECO, GridCom, ICCSEA, SCAI, UBIC, SEMIT, MLDS - 2020
pp. 01-12, 2020. CS & IT - CSCP 2020 DOI: 10.5121/csit.2020.101901

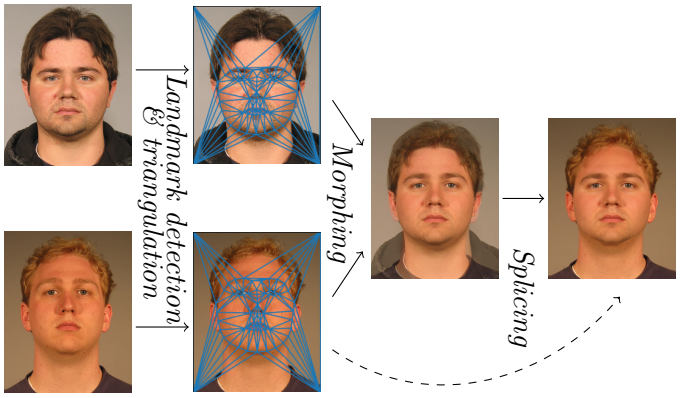


Figure 1. Morphing process

There are two scenarios in which morphing detection can take place. In a differential scenario, a second (frequently live) image of the passport holder or applicant is available for comparison. In the second, more challenging non-differential scenario, whether or not the photo has been morphed has to be decided based on the photo only.

We argue that there is a distinct possibility that carefully made morphs do not contain any artifacts that would allow a morphing detection system to distinguish them from real photos. That means we cannot rely on non-differential morphing detection to detect high-quality morphs, which leaves us with differential morphing detection methods. In that case we can use *identity*-related information to determine whether two images are of the same person. Current face recognition systems are created and trained with the purpose of verifying whether two photos are of the same person without taking into account the possibility that one of them is actually a morph. Assume we have a face recognition system (FRS) that has perfect performance on real, but not on morphed photos. When comparing two photos $X_1$ and $X_2$, if $X_1, X_2$ is a genuine pair then (a sufficient amount of) identity information in $X_1$ is also present in $X_2$, so the verification is successful. If $X_1, X_2$ is an impostor pair the identity information in $X_1$ is not present in $X_2$ and verification is unsuccessful. If $X_2$ is a morph and enough of the identity information in $X_1$ is also present in $X_2$ the verification is successful. What the FRS does not take into account is the possibility of there being identity information of a *different* person in $X_2$. The fact that many FRSs are vulnerable to morphing attacks supports this hypothesis. We argue that instead of treating face recognition and morphing detection as two separate tasks, it makes more sense to train an FRS that detects whether there are inconsistencies in identity information. This makes the task of face verification more complicated, which may lead to a lower face recognition performance, but will hopefully be better equipped to deal with the possible presence of morphs.

The rest of this paper is structured as follows: in Section 2 we will discuss related work, in Section 3 we describe our approach and in Section 4 introduce the metrics we use to evaluate our method. In Section 5 we describe how we created our dataset, which comprises both real and morphed photos. In Section 6 we describe our experiments and in Section 7 present our results. We draw some conclusions and discuss future work in Section 8.

# 2. Related Work

## 2.1. Non-Differential Morphing Attack Detection

Several methods for non-differential morphing attack detection (MAD) have been proposed. Such methods depend on finding artifacts or traces left by the morphing process to detect morphed photos. However, if a high-quality morph does not contain such artifacts or traces, then differential MAD is more suitable to address the problem. Therefore, we will not discuss non-differential MAD methods here and instead refer the reader to [2, 3] for an overview of existing methods.

## 2.2. Differential Morphing Attack Detection

*Demorphing* [8] proposes to retrieve the accomplice A's identity by subtracting an available live image from a suspected morph, but makes strong assumptions on which parameters were used for morphing. It can reduce the rate of accepted morphs, but at the cost of reducing the rate at which genuine image pairs are accepted from 99.9% to 89.2%, depending on the parameter used for demorphing. When tested on benchmark datasets in [5] equal error rates for the dection task (D-EER) of 8-16% are reported.

In [9] and [10] the locations of facial landmarks in a suspected morph are compared with the landmark locations in an available reference image. The shift between the two sets of landmark locations tends to be smaller for a pair of images with the same identity than if one of the two images is a morph. In [9] the euclidean distance and angle of the landmark shifts are used and a D-EER of 32.7% is recorded. In [10] the directed distances of the landmarks shifts are used and a spectacular D-EER of 0.00% is reported. Since the directed distances should be equivalent to using distance and angle (Cartesian vs. polar coordinates), this may indicate that some overfitting has taken place. This method of using landmark shifts achieves D-EERs of 33-39% when tested on benchmark datasets in [5].

## 2.3. Morph Attack Detection using an existing FRS

Existing face recognition systems have also been used to detect morphs. In [11], the high-level features of existing, deep-learning-based FRSs are used to train a Support Vector Machine (SVM) [12]. The resulting hyperplane is used to classify images as morphs or genuine photos. However, an SVM-based method that can separate morphs from genuine photos probably uses morphing traces and artifacts, since these are very likely to be present in an automatically created morphing dataset, and will still be present - if abstractly so - in the high-level features of an FRS. Furthermore, this method suffers from the same shortcoming, that improved MAD comes at the cost of lower genuine accept rates.

What these differential MAD methods have in common is that while they can lead to improved morph attack detection, they at the same time cause more pairs of genuine photos to be rejected, implying that the performance of face recognition on standard photos would be negatively influenced. Since we did not evaluate the performance of our method on the exact same datasets as were used in the previously mentioned publications and because our aim is to develop an FRS that is more robust to morphing attacks, whereas existing methods treat MAD and face recognition as two separate tasks, the results from other publications are not directly comparable to the results published in this paper. In practice, it might be useful for such MAD methods to be used in combination with an FRS with improved robustness to morphing attacks.

Generally, the performance of detection methods seems to vary strongly depending on the characteristics of the dataset [5].

To the best of our knowledge, no one has tried to take into account the presence of morphs during the development of an FRS.

# 3. Proposed System

## 3.1. VGG Face

The FRS we train is based on the convolutional neural network (CNN) model VGG16 that was used for face recognition in [13]. There are other FRSs that have better performance, but we chose to use this architecture since it is reasonably simple to retrain the last layer of the network, resulting in a verification system with acceptable performance with which we can perform preliminary experiments to test our hypothesis. The training method we propose can also be applied to train other (deep-learning-based) FRSs. For our experiments we resized images to $224 \times 224 \times 3$ pixels, which is the input size for the VGG16 model. Using FRSs that use larger input sizes may lead to improved performance, since more of the information contained in an image can be used.

We use the weights from a pre-trained model [14] that was trained as a classifier and only retrain the last, fully connected layer. This means that we learn a projection from the 4096-dimensional output of the pre-trained model to a 64-dimensional latent space. We train the weights $W \in \mathbb{R}^{64\times4096}$ of this last layer using the empirical *triplet loss* [15]:

$$L(W) = \sum_{(a,p,n)\in T} \max\{0, \alpha - ||x_a - x_n||_2^2 + ||x_a - x_p||_2^2\}, \tag{1}$$

where we select all possible genuine pairs $(a, p)$ and in every training epoch extend these to triplets $(a, p, n)$ by randomly selecting an image $n$ for each genuine pair such that $(a, n)$ is an impostor pair. $T$ is the set of all triplets that violate the triplet constraint:

$$\alpha + ||x_a - x_p||_2^2 < ||x_a - x_n||_2^2, \quad \alpha = 0.2. \tag{2}$$

The face embeddings $x_a, x_p, x_n \in \mathbb{R}^{64}$ are determined by forwarding the normalised output of the pre-trained network through the last, fully connected layer:

$$x_i = W\frac{f(i)}{||f(i)||}, \quad i \in \{a, p, n\}, \tag{3}$$

where $f(i) \in \mathbb{R}^{4096}$ is the output of the pre-trained network given an input image $i$. We follow the same procedure for training as in [13], and refer the reader to this publication for more details on the training procedure. Since we use a much smaller dataset for training we choose a lower latent space dimension of 64 in order to avoid overfitting.

# 4. Evaluation Metrics

This section introduces the metrics we use to measure the performance and robustness to morphing attacks of an FRS. We estimate these values using the test and validation sets.

- the EER of the face recognition system: the error rate for which the False Non-Match Rate (FNMR) and the False Match Rate (FMR) are equal: we call the threshold at which this criterion holds $t_{\text{EER}}$,
- the Morph Accept Rate at threshold $t$ (MAR($t$)): the proportion of morph pairs accepted by the FRS as a match when using a threshold $t$, where a morph pair consists of a morph and a reference image of one of the two identities present in the morph,

- the $\text{MAR}_{\text{EER}}$: the MAR at $t_{\text{EER}}$,
- the Bona fide Presentation Classification Error Rate (BPCER($t$)): the proportion of genuine pairs that are not accepted by the FRS when using a threshold $t$,
- the Attack Presentation Classification Error Rate (APCER($t$)): the proportion of (morphing) attacks that are considered a match by the FRS when using a threshold $t$,
- the EER of our differential morph attack detection (D-EER): i.e. the error rate at the threshold $t$ for which APCER($t$) = BPCER($t$),
- $\text{BPCER}_{10}$: the lowest BPCER($t$) under the condition that APCER($t$) $\leq 10\%$,
- $\text{BPCER}_{20}$: the lowest BPCER($t$) under the condition that APCER($t$) $\leq 5\%$,
- $\text{BPCER}_{100}$: the lowest BPCER($t$) under the condition that APCER($t$) $\leq 1\%$,

When using an existing FRS, the simplest way to create an MAD method would be to simply lower the decision threshold (for an FRS that uses dissimilarity scores). This provides a baseline with which the performance of other MAD methods that use features of FRSs can be compared. However, such a threshold would not be useful in practice since too many genuine claims would be rejected. Since there is often a trade-off between the performance of face verification and morphing detection [11], we display our results by plotting EER against $\text{MAR}_{\text{EER}}$. The Relative Morph Match Rate (RMMR) [16] attempts to describe something similar, but this value is rarely reported.

## 5. Creation of Morphing Dataset

We use the FRGC-dataset [17] and select the portrait-style photos, resulting in 21,772 images of 583 different identities, which we split into a training and a testing set, see Table 1. We align the images using five landmarks detected with [18] and align the faces using [19]. We crop the images using a face detector [18] and resize them to square images of 224x224 pixels.

Table 1. Our dataset.

| | # real IDs | # real imgs | # morph IDs | # morph imgs |
|---|---|---|---|---|
| Training | 514 | 19,683 | 434 | 30,924 |
| Testing | 69 | 2,089 | 99 | 4,900 |

Table 2. Validation sets.

| | # real IDs | # real imgs | # morph IDs | # morph imgs |
|---|---|---|---|---|
| PUT | 100 | 2,195 | 83 | 3,608 |
| AMSL | 102 | 204 | 1,140 | 2,175 |

The morphing method we use is based on the most-used method that consists of the following steps:

1. Landmark detection,
2. Triangulation,
3. Warping,
4. Blending,
5. Splicing (slightly different from the Poisson blending [20] that is usually used).

The morphing procedure in 1)-4) has been explained in several existing publications, to which we refer the reader for more details [1–3]. In step 5) we use a mask image to splice the inside of the morphed face into the background of one of the two original faces used to create the morph. We create this mask by using the convex hull defined by the

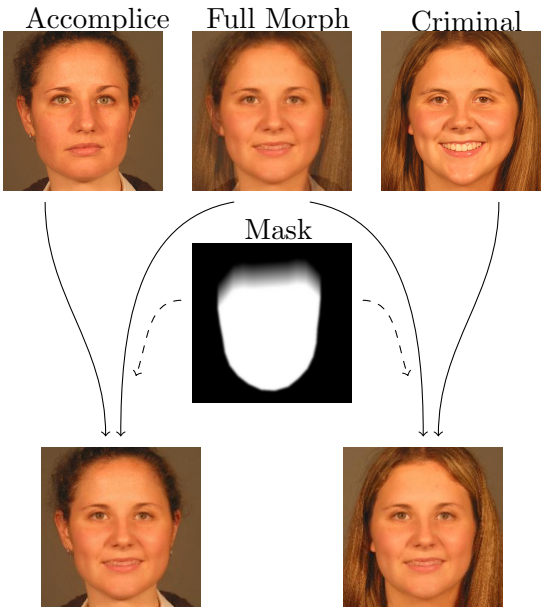


Figure 2. Splicing

outermost facial landmarks (on the jaw, chin and forehead). We ensure a smooth transition between the morph and the background on the forehead by blurring the mask in a vertical direction. We use a gaussian blur with kernel size 7x7 on the whole mask to prevent any sharp transitions, and adjust the pixel values inside the convex hull in order to ensure a natural-looking skin colour. The pixel at location $(i, j)$, $0 \leq i, j \leq 223$ in the spliced morph $M$ is

$$\begin{aligned} M(i,j) = \; & Im_1(i,j)(1 - Mask(i,j)) + \\ & (M_{\text{full}}(i,j) - \mu_M + \mu_1)Mask(i,j), \end{aligned} \tag{4}$$

where $Im_1$ is the background image into which the full morph is spliced, $Mask \in [0, 1]$ and $M_{\text{full}}$ is the full morph.

$$\mu_M = \frac{\sum_{i,j} M_{\text{full}}(i,j) \cdot Mask(i,j)}{\sum_{i,j} Mask(i,j)} \tag{5}$$

and

$$\mu_1 = \frac{\sum_{i,j} Im_1(i,j) \cdot Mask(i,j)}{\sum_{i,j} Mask(i,j)}. \tag{6}$$

See Fig. 2 for an example of the splicing step. We select pairs of identities for morphing randomly from within the training and testing set respectively, ensuring that there is no overlap in identities in the training and testing set, see Table 1. Fig. 3 shows that the majority of our morphs are accepted by two existing state of the art FRSs [18, 21].

### 5.1. Validation sets

We use two different datasets to validate our results. The first is a dataset that we created using the same pipeline as described above, but using a different dataset. For this we use the PUT Face Database [22], where we only select the subset of frontal images. For each identity $id_1$ in this dataset we determine which of the remaining identities is most similar

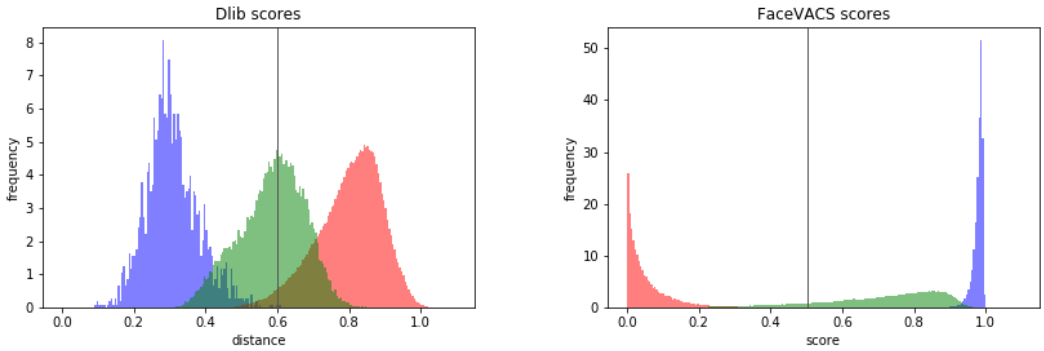


Figure 3. Evaluation of our morphed and genuine photos using two existing FRSs. The blue histograms estimate the probability density of genuine scores, the red impostor scores, and the green morph scores. Note that the dlib FRS uses dissimilarity scores whereas FaceVacs uses similarity scores. The vertical lines represent the decision thresholds recommended for these systems.

to it, which is the identity $id_2$ for which

$$\left\| \frac{1}{N_{id_1}} \sum_{i=1}^{N_{id_1}} x_i - \frac{1}{N_{id_2}} \sum_{i=1}^{N_{id_2}} y_i \right\|_2 \tag{7}$$

is minimised, where $x_i, i \in 1, ..., N_1$ are all images of $id_1$ and $y_i, i \in 1, ..., N_2$ all images of the second, to be determined, identity. The embeddings $x_i, y_j$ are computed by forwarding each image through our FRS that was trained without morphs (see Section 6.1.). We remove any duplicate pairs of identities. Since there is more pose variation in the PUT dataset, when selecting image pairs for morphing we select images that have similar poses.

The second validation set we use is the "AMSL Face Morph Image Data Set" dataset introduced in [23]. These morphs were created using images from [24] and [25].

Table 3. Training pairs.

| Types of pairs in training set | # Pairs |
|---|---|
| Genuine pairs | 592,650 |
| Genuine morph pairs | 496,494 |
| Augmented genuine pairs | 2,056,974 |

## 6. Experiments

### 6.1. Training without morphs

We follow the same procedure for training as in [13]. This means that at the beginning of every epoch a number of triplets (592,650, since this is the number of genuine pairs) is randomly generated by extending each genuine pair to a triplet as described in 3.1. We only train with the subset of triplets that violate the triplet constraint (Eq. 2). If a triplet in this subset still violates the triplet constraint at the end of an epoch, we store it and add it to the subset of triplets in the next epoch. We repeat this for a total of 200 training epochs.

### 6.2. Training with morphs

Since we use an automated pipeline to create our morphs there are some artifacts present in the morphed images. Such artifacts can be caused by badly selected landmarks, or by


Figure 4. An example of an augmented genuine, i.e. a morph of two different photos of the same person.

different expressions, see for example the mouth in Fig. 2. The morphing process also leaves some other traces, including a smoothing effect. This is caused by interpolation in the warping step, which is necessary to determine the pixel values of the warped images, and due to the blending of the two images in the splicing step. Since it is important that the FRS does not learn to separate genuine and morphed images based on such effects, and it would be very challenging to create a morphing dataset without them, instead we propose to introduce the same type of artifacts and traces in the genuine images. This can be done by creating *augmented* genuine photos. These are created in exactly the same way as morphs, but by combining two different images of the *same* person. See Fig. 4 for an example of an augmented genuine. (This technique of creating augmented genuine photos could be used as a data augmentation method in other applications, for example when there are not many images of one identity available.) See Table 3 for the number of available pairs of each type.

Table 4. Possible triplet combinations.

| $a$ | $p$ | $n$ |
|---|---|---|
| $id_1$ | $id_1$ | $id_2$ |
| $id_1$ | $id_1$ | $id_1 + id_2$ |
| $id_1 + id_2$ | $id_1 + id_2$ | $id_3$ |
| $id_1 + id_2$ | $id_1 + id_2$ | $id_1 + id_3$ |
| $id_1 + id_1$ | $id_1$ | $id_2$ |

When training with morphs, different triplet combinations $(a, p, n)$ are possible. The first possibility is that the pair $(a, p)$ can comprise two images of the same person, just as when training without morphs. A second is that either $a$ or $p$ is an augmented genuine (of the same identity), in which case we call the pair an *augmented genuine pair*. The third possibility is that $(a, p)$ consists of two morph images, both created using the same two identities. We call such pairs *genuine morph pairs*. In all three cases we extend the pair to a triplet by either selecting a third image of a different identity (with $p = 0.5$), or by selecting a morph such that one of the two identities in the morph matches that of the genuine pair. Since there are many more triplets selected at the beginning of every epoch, this means that at a fixed batch size more updates are performed in each epoch. Therefore, when we train with morphs we only train for 100 epochs. The different possible triplet combinations are summarised in Table 4, where $id_1 + id_2$ describes the identity of a morph and $id_1 + id_1$ that of an augmented genuine.

## 7. Results

Fig. 5 shows that the Equal Error Rate (EER) of an FRS decreases during both training scenarios. However, when training without morphs, after a number of updates the pro-

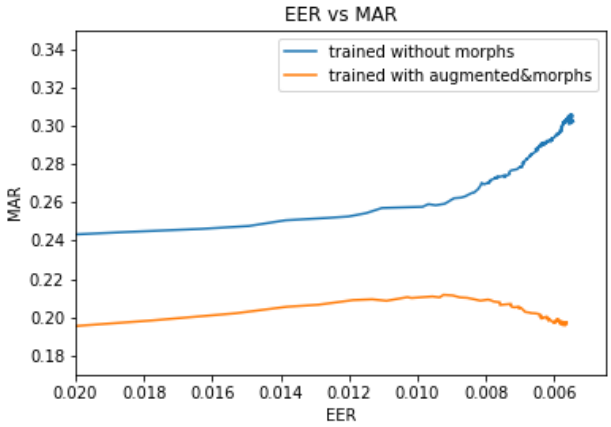


Figure 5. Performance of an FRS trained without morphs and an FRS trained with morphs and augmented genuine pairs. Results are estimated using our test set. Performance is measured using EER and $MAR_{EER}$, showing that while during training the EER decreases in both scenarios, when training without morphs this is at the cost of robustness against morphing attacks.

portion of morph pairs accepted by the FRS increases. When training with morphs, the EER also decreases, but seemingly not at the cost of accepting more morph pairs.

Table 5. Performance on test set.

| Training: | without morphs (200 epochs) | augmented&morphs (100 epochs) |
|---|---|---|
| EER (of the FRS) | 0.55% | 0.57% |
| $MAR_{EER}$ | 30.20% | 20.54% |
| D-EER | 6.70% | 5.61% |
| $BPCER_{10}$ | 4.28% | 2.41% |
| $BPCER_{20}$ | 9.38% | 6.48% |
| $BPCER_{100}$ | 28.09% | 22.50% |

In a practical verification scenario, such as border control, the thresholds at which the error rates $BPCER_{10}$, $BPCER_{20}$ and $BPCER_{100}$ are measured would not be adopted, since this would lead to the rejection of too many genuine pairs, but we report these metrics in order to allow our results to be compared to other research.

When comparing the performance on the test set to that on the PUT validation set, we no longer observe an improvement, in fact the EER when training without morphs is lower than when training with morphs while the proportions of accepted morph pairs are similar. One possible reason for this decrease in performance is that the pose variation in the PUT dataset is larger, and the validation set therefore also includes morphed images with stronger poses than were present in the training set. Since the FRS did not see such morphs during training it cannot classify them well. Another possible explanation is that the FRS has not learned to distinguish morphs from real images based on identity information, but has e.g. learned to recognise certain artifacts present in morphed images. The fact that the error rates are generally larger than on the test set indicates that this is a challenging dataset for the FRS, whether it was trained with or without morphs. Further experiments are necessary to confirm whether the lower performance on the PUT dataset

Table 6. Performance on validation sets.

| Training: / PUT: | without morphs (200 epochs) | augmented & morphs (100 epochs) |
|---|---|---|
| EER (of the FRS) | 0.69% | 0.99% |
| $MAR_{EER}$ | 83.65% | 84.62% |
| D-EER | 15.68% | 15.66% |
| $BPCER_{10}$ | 20.28% | 20.29% |
| $BPCER_{20}$ | 27.48% | 27.52% |
| $BPCER_{100}$ | 40.72% | 42.45% |
| ASML: | | |
| EER (of the FRS) | 0.00% | 0.00% |
| $MAR_{EER}$ | 24.94% | 16.48% |
| D-EER | 5.86% | 4.97% |
| $BPCER_{10}$ | 4.90% | 4.90% |
| $BPCER_{20}$ | 6.86% | 4.90% |
| $BPCER_{100}$ | 15.69% | 14.71% |

is due to the higher pose variation.

The images in the ASML dataset are quite different from the images in the training set, since the image resolution is higher and Poisson blending was used to create the spliced morphs. In spite of these differences, the FRS trained with morphs has better performance than the FRS trained without morphs. This improvement is promising, since it suggests that the FRS has indeed learned to differentiate between genuine and morph images based on identity rather than on morphing traces or artifacts.

## 8. Conclusion & Future Work

In this work we observed a modest improvement in robustness to morphing attacks after training an FRS with morphed photos. However, even a modest improvement presents an improvement on existing MAD methods, since often better detection of morphs is at the cost of decreasing performance of face recognition on normal images. We only trained the last layer of the VGG16 convolutional neural network, so more significant improvements may be achieved by training more layers of the model. Training with a larger dataset, or using data augmentation techniques are further ways to achieve better performance. Our results suggest that there may be merit to our training method, but they also underline the need for morphing databases that are more varied with respect to factors such as resolution, pose and lighting, but also variation in morphing algorithms in order to better understand which characteristics of a morphed image cause it to be challenging to classify.

Another advantage of using our method is that existing data can be used to create morphs or augmented genuines for training, which could potentially improve the performance of FRSs on normal datasets without needing to collect new data.

Finally, it is of the utmost importance that results are not only tested on types of morphs present in the training set. As we showed, these results can vary strongly when tested on different datasets.

## 9. References


[1] M. Ferrara, A. Franco, and D. Maltoni, "The magic passport," in *IEEE International Joint Conference on Biometrics*, pp. 1–7, 2014.

[2] U. Scherhag, C. Rathgeb, J. Merkle, R. Breithaupt, and C. Busch, "Face recognition systems under morphing attacks: A survey," *IEEE Access*, vol. 7, pp. 23012–23026, 2019.

[3] A. Makrushin and A. Wolf, "An overview of recent advances in assessing and mitigating the face morphing attack," in *2018 26th European Signal Processing Conference (EUSIPCO)*, pp. 1017–1021, 2018.

[4] D. Robertson, R. Kramer, and A. Burton, "Fraudulent ID using face morphs: Experiments on human and automatic recognition," *PLOS ONE*, vol. 12, p. e0173319, 03 2017.

[5] K. Raja *et al.*, "Morphing attack detection – database, evaluation platform and benchmarking," *arXiv*, 2020.

[6] L. Spreeuwers, M. Schils, and R. Veldhuis, "Towards robust evaluation of face morphing detection," in *2018 26th European Signal Processing Conference, EUSIPCO 2018*, European Signal Processing Conference, (United States), pp. 027–1031, IEEE, 11 2018.

[7] U. Scherhag, C. Rathgeb, and C. Busch, "Performance variation of morphed face image detection algorithms across different datasets," in *2018 International Workshop on Biometrics and Forensics (IWBF)*, pp. 1–6, 2018.

[8] M. Ferrara, A. Franco, and D. Maltoni, "Face demorphing," *IEEE Transactions on Information Forensics and Security*, vol. 13, no. 4, pp. 1008–1017, 2018.

[9] U. Scherhag, D. Budhrani, M. Gomez-Barrero, and C. Busch, "Detecting morphed face images using facial landmarks," in *Image and Signal Processing* (A. Mansouri, A. El Moataz, F. Nouboud, and D. Mammass, eds.), (Cham), pp. 444–452, Springer International Publishing, 2018.

[10] N. Damer, V. Boller, Y. Wainakh, F. Boutros, P. Terhörst, A. Braun, and A. Kuijper, *Detecting Face Morphing Attacks by Analyzing the Directed Distances of Facial Landmarks Shifts: 40th German Conference, GCPR 2018, Stuttgart, Germany, October 9-12, 2018, Proceedings*, pp. 518–534. 01 2019.

[11] L. Wandzik, G. Kaeding, and R. V. Garcia, "Morphing detection using a general-purpose face recognition system," in *2018 26th European Signal Processing Conference (EUSIPCO)*, pp. 1012–1016, 2018.

[12] M. A. Hearst, "Support vector machines," *IEEE Intelligent Systems*, vol. 13, pp. 18–28, jul 1998.

[13] O. M. Parkhi, A. Vedaldi, and A. Zisserman, "Deep face recognition," in *Proceedings of the British Machine Vision Conference (BMVC)* (M. W. J. Xianghua Xie and G. K. L. Tam, eds.), pp. 41.1–41.12, BMVA Press, September 2015.

[14] "VGGFace weights." https://github.com/rcmalli/keras-vggface, 2018.

[15] F. Schroff, D. Kalenichenko, and J. Philbin, "Facenet: A unified embedding for face recognition and clustering," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 815–823, 2015.

[16] U. Scherhag, A. Nautsch, C. Rathgeb, M. Gomez-Barrero, R. N. J. Veldhuis,

L. Spreeuwers, M. Schils, D. Maltoni, P. Grother, S. Marcel, R. Breithaupt, R. Ramachandra, and C. Busch, "Biometric systems under morphing attacks: Assessment of morphing techniques and vulnerability reporting," in *2017 International Conference of the Biometrics Special Interest Group (BIOSIG)*, pp. 1–7, 2017.

[17] P. J. Phillips, P. J. Flynn, T. Scruggs, K. W. Bowyer, Jin Chang, K. Hoffman, J. Marques, Jaesik Min, and W. Worek, "Overview of the face recognition grand challenge," in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, vol. 1, pp. 947–954 vol. 1, 2005.

[18] D. E. King, "Dlib-ml: A machine learning toolkit," *Journal of Machine Learning Research*, vol. 10, pp. 1755–1758, 2009.

[19] "Facealigner." https://pypi.org/project/imutils/, 2019.

[20] P. Pérez, M. Gangnet, and A. Blake, "Poisson Image Editing," *ACM Trans. Graph.*, vol. 22, p. 313–318, July 2003.

[21] "FaceVACS 9.4.0." http://www.cognitec-systems.de, 2019.

[22] A. Kasiński, A. Florek, and A. Schmidt, "The PUT face database," *Image Processing and Communications*, vol. 13, pp. 59–64, 01 2008.

[23] T. Neubert, A. Makrushin, M. Hildebrandt, C. Kraetzer, and J. Dittmann, "Extended StirTrace benchmarking of biometric and forensic qualities of morphed face images," *IET Biometrics*, vol. 7, 02 2018.

[24] L. DeBruine and B. Jones, "Face Research Lab London set," 05 2017.

[25] P. Hancock, "Psychological image collection at stirling (PICS) – 2d face sets – Utrecht ECVP." http://pics.stir.ac.uk/, 2017.