Generating facial morphs through PCA and VAE [1]

Rien Heuver[2]

Abstract— **Morphing attacks currently are a threat to face identification systems, which is why various morph detection systems are being investigated. The most-used method for morphing is the landmark-based method. Therefore, it is possible that novel morph detection systems are overfitted to detect landmark-based morphs. This research addresses methods to construct fundamentally different morphs using latent spaces. One approach uses Principal Component Analysis (PCA) for generating morphs. We found that PCA is not suitable and explain why. We also used a Variational Auto Encoder (VAE) to create a method for creating morphs through latent spaces which was more successful. The resulting morphs are not convincing enough to fool an existing face recognition system, but they are close. These VAE-based morphs were tested on an existing morph detection system, which was trained on landmark-based morphs, and it was not able to detect any of the novel morphs we created using the VAE-based method.**

I. An introduction to morphing, PCA and VAEs

Identification through facial image recognition is used in many applications, such as unlocking your phone or at border control. At border control, this process is sometimes automated. The problem is, that both the software and border officials that perform this identification can be tricked by morphed facial images[8].

A morphed facial image is a combined image of two different faces. If person A wants to fool a recognition system to believe he/she is actually person B, person A can can combine a picture of him/herself with a picture of person B. The resulting picture is called a *morph*. When an identification system looks at this morph, it will consider person A and the morph as a match, while also considering person B and the morph as a match. The most-used method for morphing images consists of landmark detection, triangulation, warping and blending. The resulting morphs are hard to detect for computers and humans alike[16].

This research attempts to find new methods of creating morphs, so new morph detection techniques have a wider spectrum of morphs to measure their performance with. The first method that has been looked into is principal component analysis (PCA). PCA is a method to achieve dimensionality reduction whilst keeping maximum information density. The other method that has been looked into is based on variational autoencoders (VAEs). VAEs are neural networks trained to encode and decode data with small information loss. Once such a neural network is trained, it can effectively achieve the same: reduce data size, while keeping as much information as possible.

[1] All code published at https://github.com/rienheuver/VAE-morpher

[2] P.R. Heuver is with Faculty of Electrical Engineering, Mathematics and Computer Science, University of Twente, 7500 AE Enschede, The Netherlands p.r.heuver at student.utwente.nl

An image can be reconstructed from the reduced data. Though their methods differ, both PCA and VAEs can perform such a reconstruction. Even though some information is lost during compression of the data, an image can be reconstructed. The more data is lost, the lesser the quality of the resulting image. We refer to the compressed data as the latent space. The general idea of morphing is then equal for both methods:

1) Compress facial images A and B into vectors in the latent space
2) Create a new latent vector by combining the vectors of A and B
3) Construct a morph by reconstructing the new latent vector

The combination of vectors A and B into a new latent vector can be done in different ways. For example, the average of the vectors A and B can be taken: $0.5 \cdot A + 0.5 \cdot B$. However, either subject could also be given a more prominent place in the resulting vector, by putting more weight on that value. This results in the following: $\alpha \cdot A + (1 - \alpha) \cdot B$, where alpha is a factor to give more prominence to one of the subjects. In both cases, experimentation is needed to see if these morphed images are realistic and if they would fool established facial recognition systems such as FaceVACS [5][1] and also see if they would fool humans.

The novel morphing techniques were evaluated using existing morph-detection systems. Since the novel methods did not exist (or at least were not widely used) when these systems were made, these systems were trained on landmark-based morphs. We therefore expect them to perform worse on detecting morphs created using our novel methods.

The remainder of this paper is structured as follows. In chapter II the research questions are outlined. An overview of the related work is summarised in chapter III. The theory of the research is further explained in chapter IV and the experiment setup on how this theory was used to answer the research questions is outlined in chapter V. In chapter VII we then proceed to interpret the results of the experiments; what do they mean and have they answered our questions? Finally in chapter VIII we address some options for future work that could improve our results.

II. Research questions

Can we use latent spaces to create convincing morphs?

1) Can PCA and VAE be used to create convincing morphs?
    a) How well can an identification system distinguish between the morphs and the two subjects of the morph?

b) How well can existing detection systems detect such morphs?

2) Are the novel morphs and more traditional morphs fundamentally different?

III. Other research in this field

A. Morph creation

Morphing an image is the process of gradually transforming one image to another and stopping somewhere along the way. The image therefore resembles both the starting and target image. The idea of using morphed face images to fool facial recognition systems stems from [8]. They manually morphed images using photo editing software and the resulting images were accepted by common facial recognition systems. This means that the facial recognition system regarded the resulting image and the original images to be images of the same person. For human operators, it would also be hard to distinguish between the morphs and genuine images. The effects of this were measured in [7], which concludes that current systems are easily fooled by manipulated images. Their suggested solution is to no longer accept images brought in by citizens during document issuing, but instead make a capture of the person in question at the moment of document issuing. However, this approach has its own drawbacks, such as no longer allowing online document requests, and realistically will not be implemented any time soon.

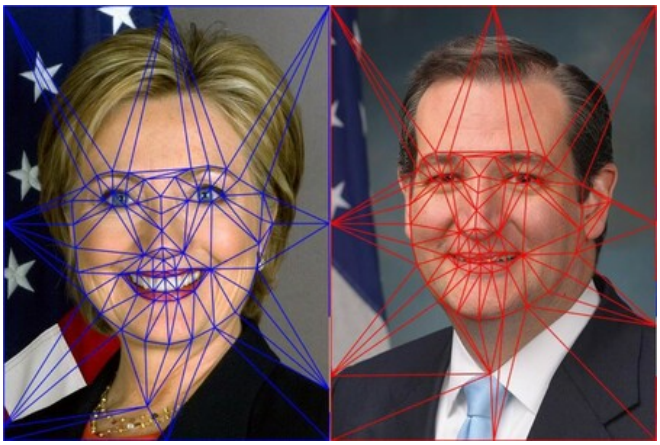

Fig. 1. Landmark detection and triangulation [17]

The most commonly used morphing procedure is based on landmarks. First, these landmarks are detected in both faces. Then these landmark-points are triangulated. See figure 1 for an example of landmark detection and triangulation. These triangles are then warped to match, after which the pixel values are interpolated for a value $0 < \alpha < 1$, where 0 means the morph will be identical to the first input image and 1 means it will be identical to the second input image. Another method is to draw corresponding lines on both images, for example around the mouth or nose. Then for each pixel the distance to each line is calculated and using these distances, corresponding pixels are found and interpolated,

Fig. 2. Morph made using landmark detection, triangulation, warping and blending [17]

creating a morphed image. The drawn lines are usually based on facial landmarks, making this method very similar to the first method. For an example of a resulting landmark-based morph, see figure 2. The procedure for creating landmark-based morphs is outlined in figure 3.

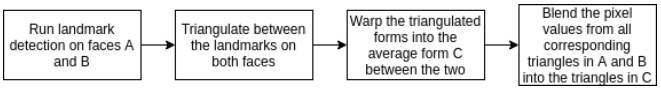

Fig. 3. Landmark-based morphing procedure

[6] used the first method and suggested that an optimal value for α is between 0.2 and 0.3. Values closer to 0.5 make it more likely that humans will not accept the morphed image if the two source faces are not very much alike.

A distinction is to be made between full morphs, as described above, and splicing morphs as described in [16]. They use the same first steps to create a morphed image, but then cut out the facial region and paste it back into one of the original images. This results in an image with fewer visible artifacts that resulted from the morphing process.

Many variations on the methods described above have been used to generate datasets of morphed images. For example, [28] and [24] use Poisson blending to improve the splicing method and further remove blending artifacts. [19] uses a combined method, taking the advantages of both complete morphing and spliced morphing.

Another method used in generating morphed images is the use of generative adversarial networks (GANs). [12] shows this on a broad level and [3] uses this technique specifically for face morphing attacks. However, thus far this has only been performed on 64x64-pixel images, so the results have no real-world applicability yet.

Many open source solutions are available to apply the above techniques, most of which use OpenCV[17].

B. Morph detection

Various methods have been investigated to detect morphed images. For example, [24] trained a Convolutional Neural Network (CNN)[15] to detect morphed images. However, they also generated the morphed images themselves, which could mean the network has been overfitted to their morphing method. [21] train a CNN on both digital and scanned pictures. They also generate their own dataset and seem to have

used rather high quality images. [20] use an SVM classifier to detect morphs after extracting Binarized Statistical Image Features (BSIF) on the images. They obtained their dataset by first taking the pictures themselves and secondly also making the morphed images from those pictures. Again, this could lead to overfitted machine learning. The third detection method [18] is based on image degradation analysis. They assume that a morphed image creates certain blending artifacts. Therefore, an authentic image will have more detectable corners in the image than morphed images.

[6] perform a technique they call 'demorphing' to attempt to retrieve the original images of the subjects in order to detect that a morph had been entered. Without their technique, a criminal has 60-70% chance to fool an automated border control (ABC) system, with their technique that lowers to 2.9-18.8%, for the best chosen α values. Their system only reports 1-2% false warnings. However, their demorphing technique assumes a morphing technique based on landmark triangulation. Therefore, it may perform much worse on different morphing techniques.

The problem with most morphing detection approaches is the underlying data. They are usually trained and/or tested on databases with one type of morph. Those that are trained on multiple types of morphs can still be fooled easily by simple image manipulation techniques, as clearly demonstrated in [26].

IV. Morph creation in the latent space

A. Normalise face pictures

Before we start training models to create morphs, we need to have a good dataset. We used the FRGC-dataset [4] and normalised the pictures from that set. We used 126 subjects for our testing set, which consists of two different images of each subject. The remaining images of those subjects and other subjects in the FRGC-dataset are used for training. This is a total of 24.332 images in the training set.

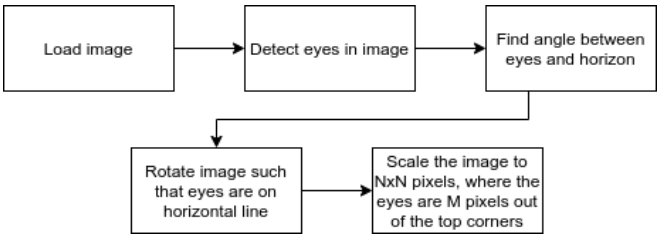

Fig. 4. Image normalisation process

For our PCA-approach we wanted the center of the eyes to always be in the exact same position, such that the biggest variance would be the differences between faces of different people. Therefore, each image goes through the procedure in figure 4. An example of a start image and normalised image using $N = 160$ and $M = 55$ is given in figure 5. These are the parameters we used for normalisation.

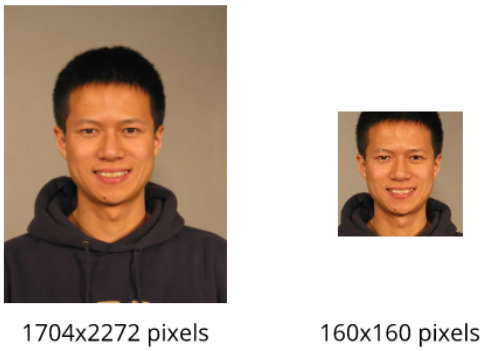

Fig. 5. Example normalisation of image

B. Principal Component Analysis

Using principal component analysis (PCA) for analysing faces was first done in [25]. The technique was later extended to use for facial recognition in [27].

A problem with using PCA for constructing images is that the result is likely to be blurry if PCA is used conventionally. [2] attempts to improve this by first reshaping the face to an average so the key facial features align better. However, this results in faces with all the same shape. A solution to this using so called eigenshapes is used in [10]. They also use PCA to generate new faces. This can be useful for example in the composition of faces from witness information. However, PCA thus far has not been used for generating face morph images.

To construct morphed images using PCA, we first went through the following training phase: take N images of $D \times D$ pixels, map each value in the range $0 - 255$ to the range $0 - 1$, then we put all pixel values for 3 colour channels, in one row of $D^2 \times 3$ values. We then have N rows of $0 - 1$ values on which we perform PCA. From that we take M principal components, also called eigenvectors or eigenfaces. After the training, we can create morphs using the following procedure: take image A and image B, map the pixel values, put them in single rows. Then project these rows on the prior chosen M principal components. This results in two latent vectors, L_A for image A and L_B for B. We then combine these two latent vectors using $L_{new} = \alpha \cdot L_A + (1 - \alpha) \cdot L_B$, where α is as explained in section I. We can now reconstruct an image by reversing the new latent vector L_{new}.

These steps are also outlined in figure 6.

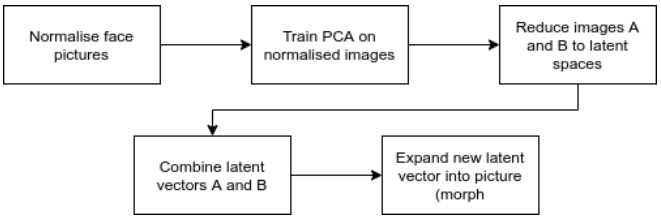

Fig. 6. PCA based morphing procedure

C. Variational Autoencoder (VAE)

The other technique we used to attempt creating morphs is called variational auto encoders [14] [22]. Auto encoders are a certain type of neural network where the dimensions of the layers of the network are large on the outside and small in the middle. The outer layers, the starting and ending layers, are of size equal to the size of the normalised images. An image is then fed through the network and some image comes out. A loss-function then determines how well the network has performed and the outcome, the loss, is used to train the network through back propagation. A properly trained network will therefore output an image comparable to the input image. However, the middle of the network consists of a layer of low dimension. Therefore, this layer contains an encoding of the image. We call this encoding the latent space of the auto encoder. An example of this is given in figure 7.

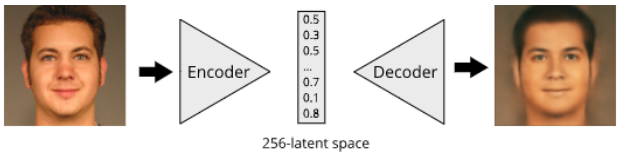

Fig. 7. VAE example

Variational auto encoders are a variation on that which tries to find a normal distribution of the latent space. Because of this normal distribution, any latent vector within that distribution will have a more predictable decoded image. By changing values in latent vectors or selecting new, random latent vectors, a VAE can be used to generate new data. In our research, VAEs are used so the average latent vector (as explained in the next paragraph) between two subjects is more likely to decode into a convincing morph.

To create morphs, we separate the left and right half of the VAE, so we can use the left half for encoding images and the right half for decoding latent vectors. Morphing is then simply done by encoding two images using the left half of the VAE into the latent space, combining their latent vectors into a new vector and then decoding this new vector using the right half of the VAE into a new image. A visual overview of this VAE-based morphing is given in figure 8.

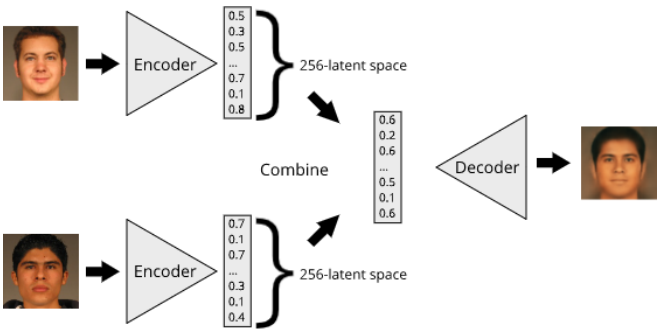

Fig. 8. VAE morphing

A variation on VAEs called β-VAE might be useful for generating morphs. This variation pushes the training of the VAE such that each node in the smallest layer, the latent space, is uncorrelated to the other nodes in that layer. Therefore, they should all learn something different about the input image. This could lead to a latent space in which each node represents a particular facial feature, therefore further separating this technique from landmark-morphing which is not based on facial features.

The steps to build a VAE-morpher are outlined in figure 9.

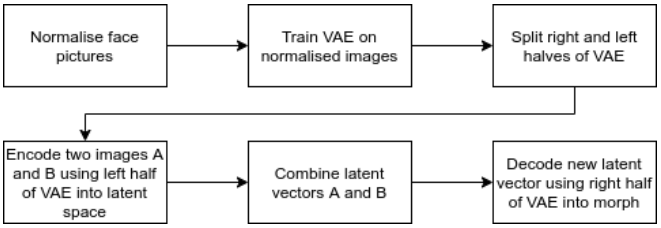

Fig. 9. VAE based morphing procedure

VAEs are able to do what they are made for, because of the structure of the neural network. However, within this structure many variations are still possible. First of all, the VAE we designed is a convolutional neural network[15], which means that we use some convolutional layers. This approach was chosen because convolutional networks have shown to be effective for networks processing images. A convolutional layer has the following parameters: kernel size, stride size, padding size and output kernels. In our network, each convolutional layer is followed up by a maxpooling-procedure[23]. Our network consists of an encoder and decoder part. The encoder consists of four convolutional layers and one fully connected layer. The decoder consists of five deconvolutional layers, that attempt to reverse the process of the encoder. Table I shows the structure for the VAE.

The fully connected layer actually consists of two layers of both 256 nodes. These are both fully connected to the output of the last convolutional layer of the encoder. The output of these fully connected layers is used for the reparameterization trick from which the latent vector is calculated. In effect, one of these two layers represents the mean and the other the standard deviation. From these two 256-length vectors, we calculate the latent vector, which is then used as input for the first layer of the decoder.

An important aspect of a VAE, and any neural network, is the loss function. We found during the training of the network that it would not capture detail very well. The output images would remain blurry. We therefore introduced an extra factor to the loss function that is intended to describe the loss in detail between the input image and the reconstruction the network makes.

1) Binary Cross Entropy (BCE) loss or reconstruction loss: this loss is a pixel-by-pixel loss between the input and output images of the network. Each epoch during training, this loss had a value between $0.55 - 0.6$.

TABLE I
LAYER STRUCTURE OF ENCODER

Encoder

Input image size	Kernel size	Stride	Padding	Output kernels	Maxpooling size	Activation function
160	4	2	2	32	2	Rectified Linear Unit
40	4	2	2	64	2	Rectified Linear Unit
10	3	2	2	128	2	Rectified Linear Unit
3	3	2	2	256	2	Rectified Linear Unit

Decoder

Input image size	Kernel size	Stride	Padding	Output kernels	Maxpooling size	Activation function
1	2	2	0	256	1	Rectified Linear Unit
2	2	2	0	128	1	Rectified Linear Unit
4	4	2	0	64	1	Rectified Linear Unit
10	4	4	0	32	1	Rectified Linear Unit
40	4	4	0	3	1	Sigmoid

Learning rate: 0.001
Optimisation method: Adam

2) Kullback Leibler Divergence (KLD) loss: this loss measures the difference between the distribution of the latent space our network has learned and a normal distribution. Each epoch during training, this loss had a value between $11,000 - 13,000$. However, we scaled this value by 10^{-6} so it would not force the network to a normal distribution too much. The result of this is that we effectively built a regular auto encoder. We tried various ways of fitting it to a normal distribution among which gradually increasing the contributing factor of this KLD-loss. However, we always found that it would either not contribute enough for the distribution to be close to normal, or it would contribute so much that all the reconstructions looked very much alike. This is further discussed in section VIII.
3) Gaussian highpass loss or detail loss: this loss is intended to measure how much detail has been lost in reconstruction. We do this by comparing a gaussian highpass version of both the input and output and measuring the distance between them. The gaussian highpass of an image is calculated by taking a gaussian blur of an image and substracting that of the image. An example is shown in figure 10. Each epoch during training, this loss had a value between $1.7 - 1.9$.

This results in the following formula: $loss = BCE + 0,000001 \cdot KLD + detail$

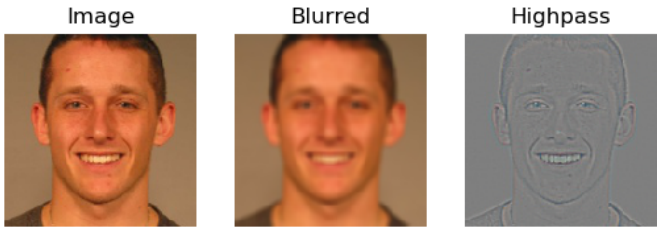

Fig. 10. Highpass image example, the right image is brightened to make the effect visual

For training the VAE, we used a learning rate of 0.001,

V. EXPERIMENTS AND RESULTS

A. Goal of the experiments

In general, the goal of the experiments is to answer the research questions. Therefore, the following experiments correspond to the aforementioned research questions in section II. To answer question 1a, we first had to build the morphing systems. The design of these systems is further explained in sections V-B.1 and V-B.2. We then performed an experiment to test them which is addressed in V-B.5. Question 1b is addressed in section V-B.3. Question 2 is partly answered by section V-B.3 and partly by section V-B.4. Table II visualises which experiment is intended to answer which research question.

TABLE II
RESEARCH QUESTIONS AND EXPERIMENTS

Research question / Experiment	V-B.5	V-B.3	V-B.4
1a	x		
1b		x	
2		x	x

All source code produced for this research is open sourced at [11].

B. Setup of the experiments

1) Build morpher based on PCA: We found that creating morphs through PCA does not work. The best image that can be created with PCA is exactly equal to adding up the two source images and dividing by 2. This is in a situation where all information is retained, which is not the goal of PCA. This is because all PCA operations are linear operations. The problem is shown in figure 11 where all blocks with an apostrophe indicate latent-vectors. In the upper half of the image we see the intended procedure for creating a morph through PCA. In the bottom half however, we see what

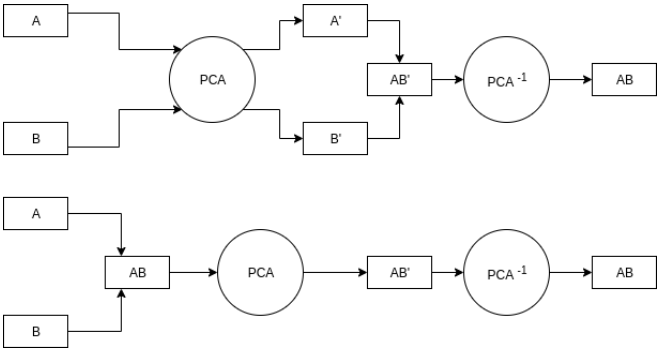

Fig. 11. PCA problem

happens if we first add up the two source images and then put it through the same procedure. The result is the exact same image.

2) Training the morpher based on VAE: After training our VAE-model for quite a while, we can the reconstructions it makes of the input images look like the input image, as intended. Some images however, are too hard for it to accurately reconstruct, because of posture, hair or other variations that are relatively scarce in the dataset. In figure 12 we see a reconstruction that is not accurate and in figure 13 we see one that was reconstructed seemingly better.

Fig. 12. Relatively bad reconstruction by VAE

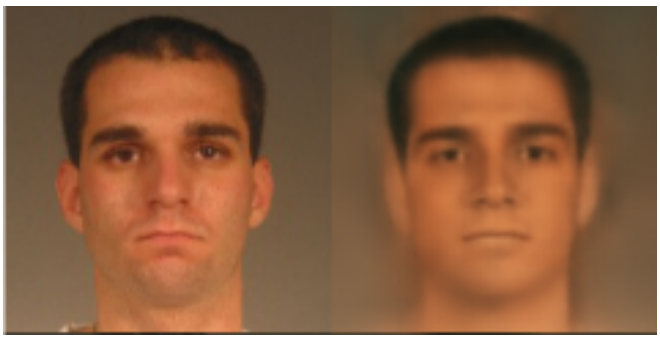

Fig. 13. Relatively good reconstruction by VAE

In both images we can see that, even though we use the detail loss, it has difficulty reconstructing detail in the image. The reconstructions are always blurry compared to the input image. Since the model only seems to be able to reconstruct blurry-looking images, the morphs are also blurry. In figure 14 we see a few morphs generated by our model using the earlier described methods.

Fig. 14. From left to right: input A, reconstruction of A, morph, reconstruction of B, input B

3) Test morphs on existing detector: The goal of this research is not only to create convincing morphs, which is addressed in the next section, but also to generate morphs that are fundamentally different from morphs created using the conventional triangulation approach. If our morphs are not detected by a detection system that is effective in

detecting triangulation morphs, then our morphs are therefore fundamentally different.

We tested our morphs on an existing identifier that is based on detecting local binary patterns (LBPs). To train this system we generated landmark-based morphs with the same dataset, including normalisation, as that we used for our VAE. The detection results are shown in III. The detector was able to detect 100% of the landmark-based morphs during testing, but was unable to detect any of our novel morphs.

TABLE III
MORPH DETECTION SCORES

	Detected
Landmark morphs	100%
VAE morphs	0%

4) Difference between morphing methods: Another way of measuring how different our novel morphs are from the landmark-based morphs is by measuring the distance between the two. By distance, we mean the measured distance by a face recognition system. To test this, we used a python implementation[9] of a face recognition model[13] by dlib. The method is as follows: take pictures of two different subjects $SubjectA$ and $SubjectB$. Create a landmark-based morph from these two images and create a novel morph from these two images. Then measure the distance between these two morphs. If the distance is small, it means the morphs are, in the eyes of the face recogniser, very alike. If the distance is high, it means they are different.

In figure 15 we see an example of two subjects and morphs created with the two different methods.

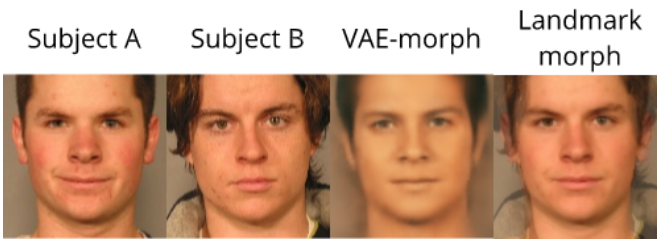

Fig. 15. An example of the different morphing methods

We measured the distances for the two morphing methods for a total of 249 morphs. The distances between these two morphs are plotted in figure 16. This figure shows that, from the perspective of the identifier, the two morphing systems are not very alike. The orange part is for combinations of subjects for which the VAE-morph was accepted as both subject A and B. We see that these successful morphs are more like the landmark-based morphs than when we look at all morphs combined, but are still quite different.

5) Test morphs with source images on existing identifier: If our morphs are not convincing, they are of no use. Therefore, we want to know whether or not our morphs can be used to fool a face recognizer. For this we used the one as

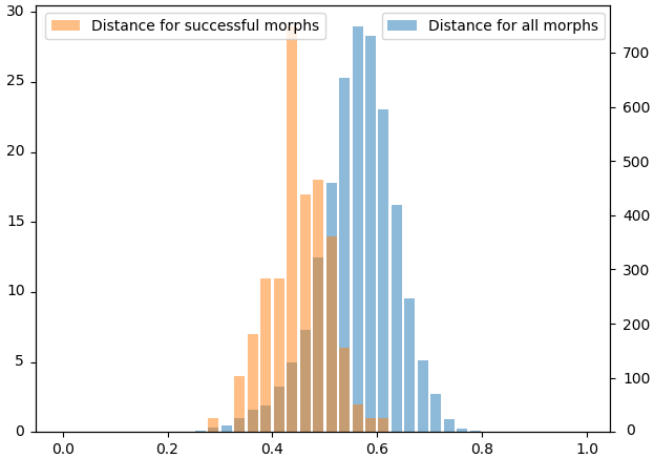

Fig. 16. Distances between the novel and landmark-based morphing methods

described in section V-B.4. To run this experiment we need to have two different pictures of two different subjects. We call these subjects $SubjectA$ and $SubjectB$ and there respective samples A_1, A_2, B_1 and B_2. Samples A_1 and B_1 get passed through our network resulting in latent vectors (L_{A1}, L_{B1}) and a reconstruction (R_{A1}, R_{B1}). To create a morph from samples A_1 and B_1, we calculate $L_{morph} = \frac{L_{A1}+L_{B1}}{2}$. That morph is then passed through the decoder part of the network, resulting in a morph.

To see how well our morphs work, we measure various distances between faces. This is done by encoding the faces in two vectors of size 128 and measuring the euclidean distance between those two vectors. The distance, $0 - 1$, is a metric for how different two faces are, supplied by the face recognition system. A low number means two faces are likely from the same subject, a high number means they are different. The face recognition model uses a threshold of 0.6, meaning that anything below that threshold indicates two faces belong to the same subject.

For a morph to fool the recognition system, this means that the distance between the morph and samples A_2 and B_2 should be below the threshold. Besides that, we also measure the genuine and imposter scores:

- Sample 1, sample 2: this indicates how different the two sample images of the subject are. This is also the distance that is measured with regular uses of an identifier: it compares a live image of a subject with a reference picture. This distance is commonly known as the genuine score.
- Sample 2, morph: this indicates how much the morph and the subjects are alike. The goal of the morphs is to make this distance close to the genuine score. We use sample 2 to resemble a situation where a live image is compared to a morph reference picture. We call this score the morph score.
- Sample 1 A, sample 2 B: this is the distance between the two different subjects. I.e. the situation where subject B tries to identify as subject A. This is commonly known

as the imposter score.

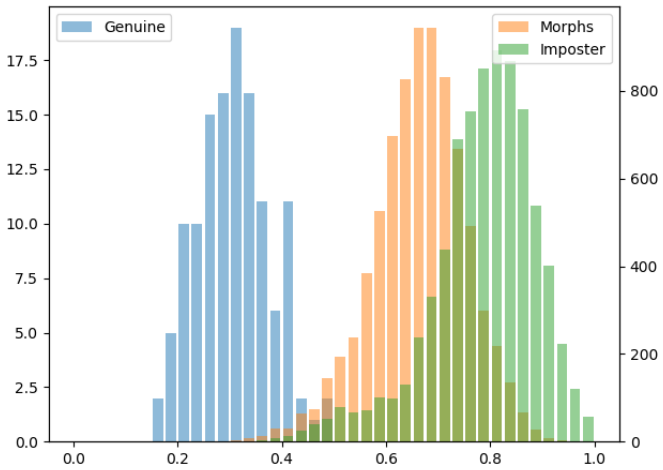

Fig. 17. Genuine, morph and imposter scores for the novel method

In figure 17 we see the genuine, morph and imposter scores. The y-axis is labeled twice, since there are only 126 subjects and thus genuine scores, whilst there are $\frac{126*125}{2}$ combinations of subjects and thus morph and imposter scores. A perfect identification system will have the genuine and imposter scores completely separated. If the morph scores are equal to the imposter scores, it means they are not working. If they are equal to the genuine scores they are working. We can see that the morphs are in the middle, but more on the imposter side of the scores. However, we can also see that some morphs certainly overlap with the genuine scores, indicating that some successful morphs are present.

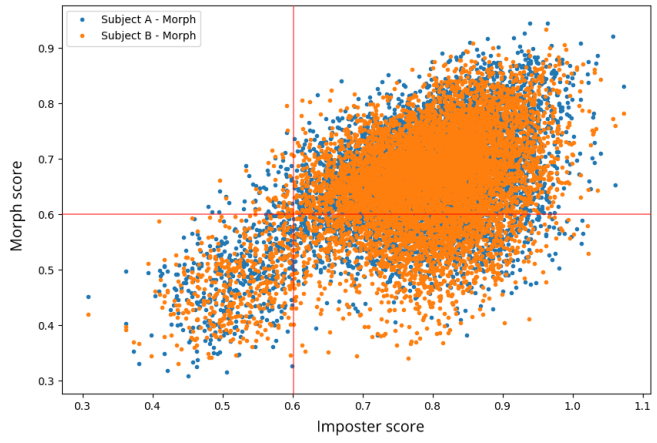

Fig. 18. Morph scores and imposter scores plotted

To further examine these scores, we can look at figure 18. In this graph we can see lots of morphs plotted. On the X-axis is the distance between subject A and B; the imposter score. On the Y-axis is the distance between the morph and either subject A (orange) or B (blue); the morph score. The lines indicate the threshold of the face identification system. Thus, all successful morphs are below the horizontal line.

If a morph is successful, it does not imply that it also looks convincing to the human eye. In figure 19 we see an

Fig. 19. Odd looking, but successful morph between a man and a woman

example of a successful morph. In this case we morphed a man and a woman and the resulting morph is accepted for both subjects, but to the human eye this is not convincing at all.

Fig. 20. Successful morph that looks better

However, in figure 20 we see another successful morph, but one that looks much better.

Fig. 21. Successful morph between two men that look alike

In figure 21 we see an example of two men that look alike. The distance between the subjects' original images is already below the threshold. The resulting morph is as well for both subjects.

VI. Discussions

In figure 17 we see the distribution of our three scores: genuine, morph and imposter. The figure shows that most of the morphs are below the threshold of 0.6, but that quite a lot are also below that threshold. This indicates that although the morphs are not successful on average, we have generated there are successful morphs, such as figures 19 and 20.

We can also see in figure 17 that the distribution of the morphs is shifted to the left compared to the imposter scores. This indicates that overall, our morphs do improve the scores and thus make it harder for a face recognition system.

In figure 18 we see a clear pattern from the bottom left to the top right. This means that the less subject A and B look like each other, the less the resulting morph will look like A and B and vice versa. If subject A and B do not look like each

other, their imposter score will be higher than 0.6. I.e. most imposter scores will be on the right side of the vertical line. If a morph is successful, the distance to it and subject A or B will be lower than 0.6, thus be below the horizontal line. The morphs that therefore truly fool the system are in the lower right quadrant: morphs that are below the threshold whereas the distance between the subjects is above the threshold.

This analysis of our morphs gives us an answer to research question 1 and its sub-questions: yes we can create convincing morphs. For a criminal to be successful, he/she does not need all morphs or even for the majority of them to be successful; just one successful morph suffices.

To answer research question 2 we added another experiment in section V-B.4. In that experiment we use the face identifier to measure the distance between the landmark-morph and the novel morph of the same subjects A and B. The distribution of that distance in figure 16, shows that the morphs are clearly different. Quite some of these morphs are so different that the identifier regards them as different people, seeing their distance is above the 0.6 threshold. This, combined with the experiment in section V-B.3 gives us an answer to question 2: yes they are fundamentally different.

VII. Conclusions

In this research we have tried to answer the following questions:

1) Can PCA and VAE be used to create convincing morphs?
    a) How well can an identification system distinguish between the morphs and the two subjects of the morph?
    b) How well can existing detection systems detect such morphs?
2) Are the novel morphs and more traditional morphs fundamentally different?

In order to answer these questions, we built two morphing-systems; one based on Principal Component Analysis (PCA) and one based on Variational Auto Encoders (VAEs). The PCA-based morphing system turned out to be impossible to make, therefore answering it's respective half of question 1.

The VAE-based system yielded good results as we have seen in the experiments and discussions. We built a system based on a convolutional neural network with in the middle of the network a small, fully connected layer, resulting in a latent space. The left half of the network encodes the image into the latent space, whereas the right half decodes the latent space into an image. Using these halves separately, we can encode two different images, average their latent spaces and decode the result into a new image. That resulting image is then a morph of the input images.

The analysis and discussion of the distances, as presented in section VI gives us an answer to research question 1a: yes, it is possible to find a morph between two subjects that the identifier will recognize as both subjects.

The experiment in section V-B.3 shows us that these morphs are truly different from the landmark-based morphs and that an existing detection system, which successfully detects landmark-morphs, cannot detect our novel morphs. This gives us an answer to research question 1b.

In conclusion, this means that VAEs can indeed be used to create convincing morphs, therefore answering question 1.

To be useful, these morphs should also prove to be fundamentally different from landmark-based morphs. Because if not, why go throught the effort of generating such morphs if the landmark-based method holds the same result. In the discussions we saw two experiments that show that the novel morphs are indeed fundamentally different from the landmark-based morphs. Giving research question 2 the answer: yes.

VIII. Future work

We tried a number of variations to improve our results and thus our morphs. However, more variations can still be looked into that could hold better results. A few suggestions are following.

1) Different structures for the VAE: In our research we tried used convolutional layers with certain parameters. Tweaking these parameters or adding more layers might hold different results. For example, it's possible that our max-pooling procedure loses too much information which might in turn be part of the cause our reconstructions are blurry. Experimenting with different structures could make a difference.

Another improvement in this regard could be to train on higher resolution images combined with a bigger latent space. We first trained our model on a smaller latent space which resulted in reconstructions with even less detail than we have now.

As discussed in section IV-C, the network we eventually trained and tested is effectively a regular auto encoder and not a variational auto encoder. We attempted different methods of implementing the variational part:

1) Find a static factor for the KLD-loss such that a normal distribution is approached by the network while still having good reconstructions (low BCE-score).
2) First train the network without the KLD-loss, but once the network has learned to make good reconstructions, add in the KLD-loss.
3) Gradually build up the KLD-loss by multiplying the KLD-loss with the epoch number.

None of the above methods yielded good results for our research. However, other methods could be attempted to better introduce the variational part of the VAE.

2) Initialising the network with an average face: In theory, the purpose of the model is to learn what makes subjects' faces unique and capture that in the latent space such that it can reconstruct it. To that extend, initialising the weights

of the neural network in such a way that it will reconstruct an average face, could help the training of the network focus more on what makes subjects' faces unique and could in turn improve the amount of detail in the reconstructions.

3) Different datasets: In this research we use a large dataset with 126 different subjects. The images in the dataset are normalised up to a certain degree. However, variations in posture and emotional expression are still present and could cause too much variation for the model to learn. To make a convincing morph between two subjects, it does not need to learn all this variation. Therefore, adjusting the dataset to contain homogeneous facial expressions and postures could improve morphing results, whilst having no effect on the quality of the morph.

4) β-VAE: In section IV-C we discussed the option of using a variation on VAEs called β-VAE. In this research we did test this variation, so it is still worth trying to see if that yields better morphs.

REFERENCES

[1] *Cognitec's Best-in-class Facial Recognition Solutions Honored by Frost & Sullivan*. https://markets.businessinsider.com/news/stocks/cognitec-s-best-in-class-facial-recognition-solutions-honored-by-frost-sullivan-1028048075. [Online; accessed 04-09-2019]. 2019.

[2] Ian Craw and Peter Cameron. "Parameterising images for recognition and reconstruction". In: *BMVC91*. Springer, 1991, pp. 367–370.

[3] Naser Damer et al. "MorGAN: Recognition vulnerability and attack detectability of face morphing attacks created by generative adversarial network". In: *2018 IEEE 9th International Conference on Biometrics Theory, Applications and Systems (BTAS)*. IEEE. 2018, pp. 1–10.

[4] *Face Recognition Grand Challenge data set*. https://www.nist.gov/programs-projects/face-recognition-grand-challenge-frgc. [Online; accessed 08-10-2019]. 2010.

[5] *FaceVACS-Entry*.

[6] Matteo Ferrara, Annalisa Franco, and Davide Maltoni. "Face demorphing". In: *IEEE Transactions on Information Forensics and Security* 13.4 (2017), pp. 1008–1017.

[7] Matteo Ferrara, Annalisa Franco, and Davide Maltoni. "On the effects of image alterations on face recognition accuracy". In: *Face recognition across the imaging Spectrum*. Springer, 2016, pp. 195–222.

[8] Matteo Ferrara, Annalisa Franco, and Davide Maltoni. "The magic passport". In: *IEEE International Joint Conference on Biometrics*. IEEE. 2014, pp. 1–7.

[9] Adam Geitgey. *Face Recognition*. https://github.com/ageitgey/face_recognition. [Online; accessed 28-april-2020].

[10] Peter JB Hancock. "Evolving faces from principal components". In: *Behavior Research Methods, Instruments, & Computers* 32.2 (2000), pp. 327–333.

[11] Rien Heuver. *VAE-morpher*. https://github.com/rienheuver/VAE-morpher.

[12] Tero Karras et al. "Progressive growing of gans for improved quality, stability, and variation". In: *arXiv preprint arXiv:1710.10196* (2017).

[13] Davis King. *High Quality Face Recognition with Deep Metric Learning*. http://blog.dlib.net/2017/02/high-quality-face-recognition-with-deep.html. [Online; accessed 28-april-2020]. 2017.

[14] Diederik P Kingma and Max Welling. "Auto-encoding variational bayes". In: *arXiv preprint arXiv:1312.6114* (2013).

[15] Yann LeCun et al. "Object recognition with gradient-based learning". In: *Shape, contour and grouping in computer vision*. Springer, 1999, pp. 319–345.

[16] Andrey Makrushin, Tom Neubert, and Jana Dittmann. "Automatic Generation and Detection of Visually Faultless Facial Morphs." In: *VISIGRAPP (6: VISAPP)*. 2017, pp. 39–50.

[17] Satya Mallick. *Face Morph using OpenCV - C++ / Python*. https://www.learnopencv.com/face-morph-using-opencv-cpp-python/. [Online; accessed 25-June-2019]. 2016.

[18] Tom Neubert. "Face morphing detection: An approach based on image degradation analysis". In: *International Workshop on Digital Watermarking*. Springer. 2017, pp. 93–106.

[19] Tom Neubert et al. "Extended stirtrace benchmarking of biometric and forensic qualities of morphed face images". In: *IET Biometrics* 7.4 (2018), pp. 325–332.

[20] R Raghavendra, Kiran B. Raja, and Christoph Busch. "Detecting Morphed Face Images". In: Sept. 2016. DOI: 10.1109/BTAS.2016.7791169.

[21] R Raghavendra et al. "Transferable deep-cnn features for detecting digital and print-scanned morphed face images". In: *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. IEEE. 2017, pp. 1822–1830.

[22] Danilo Jimenez Rezende, Shakir Mohamed, and Daan Wierstra. "Stochastic backpropagation and approximate inference in deep generative models". In: *arXiv preprint arXiv:1401.4082* (2014).

[23] Dominik Scherer, Andreas Müller, and Sven Behnke. "Evaluation of Pooling Operations in Convolutional Architectures for Object Recognition". In: *Artificial Neural Networks – ICANN 2010*. Ed. by Konstantinos Diamantaras, Wlodek Duch, and Lazaros S. Iliadis. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 92–101. ISBN: 978-3-642-15825-4.

[24] Clemens Seibold et al. "Detection of face morphing attacks by deep learning". In: *International Workshop on Digital Watermarking*. Springer. 2017, pp. 107–120.

[25] Lawrence Sirovich and Michael Kirby. “Low-dimensional procedure for the characterization of human faces”. In: *Josa a* 4.3 (1987), pp. 519–524.

[26] Luuk Spreeuwers, Maikel Schils, and Raymond Veldhuis. “Towards robust evaluation of face morphing detection”. In: *2018 26th European Signal Processing Conference (EUSIPCO)*. IEEE. 2018, pp. 1027–1031.

[27] Matthew A Turk and Alex P Pentland. “Face recognition using eigenfaces”. In: *Proceedings. 1991 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. IEEE. 1991, pp. 586–591.

[28] Lukasz Wandzik et al. “CNNs under attack: on the vulnerability of deep neural networks based face recognition to image morphing”. In: *International Workshop on Digital Watermarking*. Springer. 2017, pp. 121–135.