Engineering and Technology Journal e-ISSN: 2456-3358
Volume 09 Issue 11 November-2024, Page No.- 5531-5543
DOI: 10.47191/etj/v9i11.12, I.F. – 8.227

Exploring the Effectiveness of Different Morphed Face Generation Techniques: A Comparative Analysis

Ghadeer I. Maki
Technical Institute of Al-Diwaniyah, Al-Furat Al-Awsat Tehnical University (ATU), Al-Diwaniyah, Iraq

ABSTRACT: Generating morphed faces has become an important area of research with various applications. Traditional morphing techniques have limitations in accuracy and realism, especially when dealing with complex facial expressions and identities. This paper compares morphed face-generation techniques, including conventional morphing and deep learning-based approaches. We discuss the strengths and weaknesses of each method, highlighting the limitations of traditional techniques and the promise of deep learning-based techniques in generating highly realistic and diverse morphed faces. We also explore recent advances in deep learning-based morphing techniques, including style transfer and attention mechanisms, which provide more fine-grained control over the generated output. Finally, we discuss the challenges that need to be addressed in this field, such as the need for large amounts of training data and ethical considerations. The continued development of morphed face-generation techniques will likely lead to exciting new applications in various fields, including entertainment, advertising, and social media.

KEYWORDS: Semi-Manual morph, GAN, StyleGAN, Wavelet approach, VAE.

1. INTRODUCTION

There are a variety of applications for facial image recognition for identification, including being able to access your phone and control entry into a country. This procedure is occasionally carried out by an automatic system at the port of entry. The issue is that morphed facial images can fool the software and the people who work at the border gate who do this identification process. A morphed face is an image of two people's features superimposed on each other. Meaning is the combination of two distinct facial images [1]. The process of gradually morphing one picture into another and pausing at some point along the way is what we mean when we talk about morphing an image. As a result, the picture is comparable to both the starting image and the target image. Using images of morphed faces to trick facial recognition systems (FRS) originated here. They used image editing software to morph images physically, and FRS accepted the resulting images. This indicates that the facial recognition system considered the resulting image and the original images to be pictures of the same individual because they shared similar characteristics [2]. It would also be difficult for human operators to differentiate between the morphs and the actual images. After measuring its effects, it was determined that existing systems are easily duped by pictures that have been tampered with [3]. If individual A wishes to trick a recognition system into thinking it is B, individual A can combine a picture of themselves with a photo of B. This will cause the system to believe that person A is person B. A morph is a name given to the image that is produced. When looking at this morph, an identification algorithm will consider person A and the morph to be a match, and it will also consider person B and the morph to be a match. The morphs produced by some methods are challenging to identify for both computers and people [4]. The morphing face has excellent potential for optical as well as electronic illusion. This research presents ways of producing morphs that provide an electronic illusion for developed morph detection strategies. The techniques used to generate a morphed can be divided into two categories: deep learning and machine learning. Each of these methods has advantages and disadvantages that are reflected in the generated images or employed technology.

The paper's mentioned basic morphing procedures can be carried out by various commercial software or programs. Most of these programs are highly user-easily, and it is possible to generate a morphing sequence between two images briefly. In this paper, I will discuss the morphing algorithm used in some software packages and explain how these algorithms solve problems that most software cannot, such as morphing only a portion of an image while leaving the rest unchanged or creating a morphed image without artifacts.

2. MORPH CREATION

Morphing has traditionally been employed in the animation industry to produce cinematic special effects. (Wolberg, 1990). The process of specifying a metamorphosis in image processing such that one image can be transformed into another is referred to as "morphing," The term "morphing" is used to denote the approach. For instance, if there are two photographs, such as A and B, it is possible to generate new images that change A into B and vice versa. The position of the generated image along the continuum from A to B can be determined parametrically. The morphing algorithm can create an image at any point along this continuum. Because it enables the specification of the proportion of "A" and "B" in the created image, parametrization is a valuable feature for generating images for psychological studies. This is because it enables the similarity of the generated image to be adjusted about A and B. Calculating a morph between points A and B may be broken down into two independent phases: warping and cross-dissolving. First, picture A is warped to image A', and image B is warped to image B'. This implies that the shapes of image A and image B are deformed toward A' and B', respectively, so A' and B' have similar forms. After that, images A' and B' are cross-dissolved, which means that the colors or grayscale values of images A' and B' are combined to generate a new image. This unique image can share images A and B's shape and color characteristics [4,5].

3. Face Morphing Techniques

In this section, we will describe the most prevalent morphing techniques. Generally, the methods for generating morphed images can be categorized as (a) Landmark based and (b) Deep Learning-based. Figure 1 illustrates these techniques.

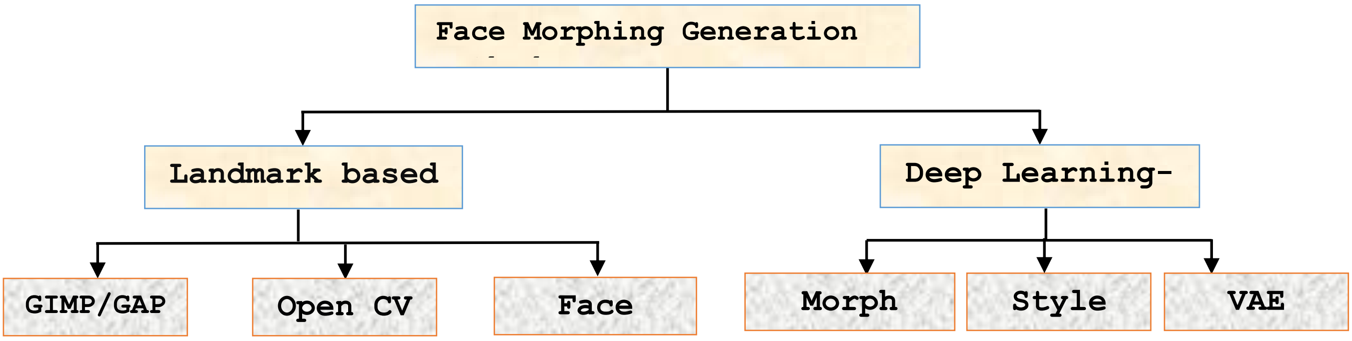


Figure 1. Explain morphing

3.1 Generation Morph Images Based on Landmarks

Obtaining the landmark points on the facial region, such as the nose, eye, and mouth region, is the first step in the landmark-based morph generation process [6]. The pixels are moved to various positions that are more averaged out to warp these landmark points gathered from both faces. To execute warping, move the pixel positions of both contributing subjects to the point closest to the landmark. Later, the idea of Delaunay triangulation was used, which involved warping and moving the pixels of both contributing facial photos in various directions to form triangles. When combining images (that will be morphed), the blending or morphing factors should be considered. The morphing factor of 0.5 is utilized by face morphing apps to develop morphs of high quality and practicality that can resemble both contributing subjects to an equal degree. Because the morphing process involves replacing the pixel positions, there is a possibility that some of the pixels will not be appropriately aligned. This might result in noise-generating artifacts and ghostlike images, giving the impression that the images are not authentic. (i.e., easy to detect by the human observer). Therefore, to increase the brightness and contrast of the image, it is customary to carry out several postprocessing stages. These procedures include image smoothing, sharpening, edge correction, histogram equalization, manual retouching, and image enhancement. These steps can decrease or minimize the artifacts produced during the morphing process. Landmarks methods are divided into two parts manual and automatic. The images requiring manual intervention are characterized by identifying the face landmarks for the two individuals whose faces are to be merged, as shown in Figure 2. Regarding the images, they are automatic and do not require user interaction [6, 7].

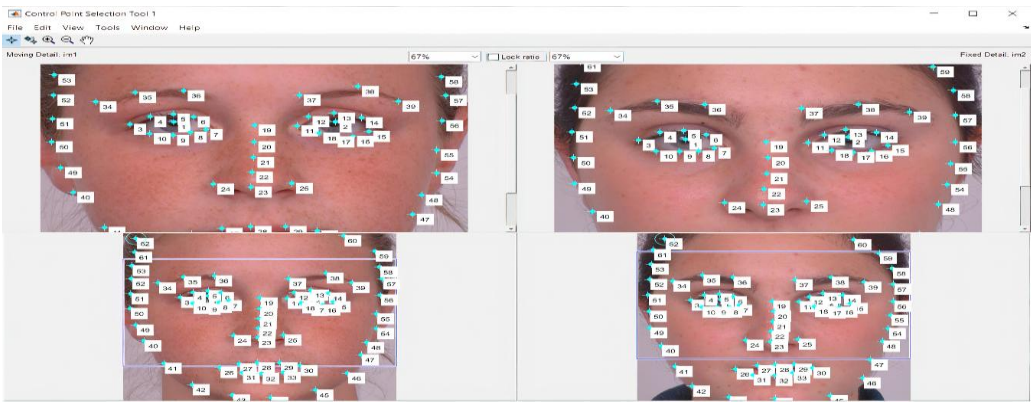


Figure 2. Select manual landmarks for morphing.

The process of morphing does not generate a single image but rather a series of images (a collection of frames) that comprise a video. The initial image of person A is merged with the characteristics of person B until the final image, which represents person B.

3.1.1 Manual Landmarks: a method based on manually selecting landmarks at the project's outset. This method comprises the following steps:

1. Read the actual image for person A.
2. Read the actual image for person B.
3. Determine the total number of frames in the video and the duration between each frame.
4. Determine the extension of images (frames).
5. The image is converted into a digital matrix to make it easier to deal with.
6. using the Viola-Jones algorithm, determine only the face region that includes the nose, mouth, eyes, and eyebrows.
7. After the preceding stage, the face is extracted from a new image and worked on.
8. Define pairs of corresponding points. The two images are displayed, and then the selection of the facial landmarks of the two images is made manually and sequentially, such that each point determined in the first image is at the exact location or near to it in the second image, as illustrated in Figure 1.
9. The mean of the two-point sets computes the triangulation to lessen the possible triangle deformations.
10. Implementing a warping process between A and B by steps 8 and 9.
11. Obtain a new shape from the original images using defined point correspondences.

The unrestricted selection of the features to be combined and the estimation of the warping fraction are advantages of this method. When the conditions and motion of the two images are similar, the results are favorable.

3.1.2 Automatic Landmarks: For identifying landmarks automatically, using three types of programs [8, 9, 10]:

A. GIMP/ GAP: abbreviation "GIMP" stands for "GNU Image Manipulation Program." This tool can be downloaded for free and used for various picture editing activities, including editing, authoring, composition, and retouching images.

B. Face Fusion: Numerous commercial programs on the Internet are available for free or for a fee that can generate morph images. All of these programs utilize the landmarks-based methodology, and these programs include the following:

1. Morph Thing
2. 3Dthis Face Morph
3. Face Swap Online
4. Abrosoft Fanta Morph
5. Face Morpher
6. Magic Morph

The images produced by these programs are of poor quality and contain anomalies that require manual removal. In the results section, you can observe the output images of these programs.

C. Open CV: The method is a modification of an open-source implementation that is utilized for morphing faces by making use of the 68-point annotator that is located in the Dlib library. These landmarks can be found in the regions surrounding the mouth, eyebrows, eyes, nose, and the face's boundary, as shown in Figure 3. After obtaining face landmarks for each of the two genuine source photos, these landmarks form Delaunay triangles, which are afterward warped and blended based on alpha.

The concept of morphing is easily understood. We have two images, A and B, and we want to produce an image N that is in between the two via a blending operation. A parameter called 'alpha' with a value between 0 and 1 ($0 \leq \alpha \leq 1$) controls how images A and B blend. The morph N takes on the appearance of A when alpha is set to 0, but when alpha is set to 1, N takes on the appearance of B. Put, you can blend the images by applying the equation (r, c) to each pixel in the image [6, 7].

Figure 3. Explain 68 points for determining landmarks.

$$N(r,c) = (1-\alpha)A(r,c) + \alpha B(r,c) \qquad (1)$$

Therefore, the first step in morphing image A into image B is determining the pixel correspondence between A and B. In other words, we need to locate the pixel in image A (r,c) corresponding to each pixel B (r,c). This can be done by comparing the two images side-by-side. We can mix the images in only two stages when using alignment to find these correspondences. To begin, we have to perform the calculations necessary to determine the pixel's position (r_N, c_N) In the morphed image. The following mathematical expression describes it:

$$r_N = (1-\alpha)r + \alpha r \qquad (2)$$

$$c_N = (1-\alpha)c + \alpha c \qquad (3)$$

We must determine the pixel's intensity at (r_N, c_N) Using the following formula:

$$N(r_N, c_N) = (1-\alpha)A(r,c) + \alpha B(r,c) \qquad (4)$$

An open CV consists of the following steps:

1. Landmark Detection

Finding point correspondences is a strategy that can get a significant number of points by automatically detecting face feature points, as shown in Figure 3. Found 68 matching points by using the dlib library. After that, I placed one more point on the right side of the ear on the left side of the head. One can either increase the number of points around the head and neck to achieve even better results or remove the manually selected points to obtain results that are slightly less accurate but entirely automatic.

2. Triangulation

Triangulation is utilized in various applications, such as face morphing, face replacement, etc. The previous stage yielded two groups of 80 points, one for each image (A, B). Therefore, determine the average of the matching points in the two groups to arrive at a group of 80 points. The outcome of Delaunay triangulation is producing a list of triangles, each represented by an index from the array of 80 points. In this instance, triangulation results in 149 triangles joining the 80 points. The triangulation is kept as an array with three columns for storage. as shown in Figure 4.

Figure 4. Triangulation of face using Delaunay.

3. Warping

This stage can be divided into three fundamental steps:

a) In a morphed image, one must identify the positions of feature points using equations (2, 3). We can identify the locations of all 80 points in the morphed image M. The coordinates denote these locations r_N and c_N.

b) computed affine transforms: 80 points for each from A, B, and morphed image. Triangulation defined in stage 2 calculated the affine transform that maps the triangle's three corners in A to the three corners of the triangle in the morphed image, given a triangle in A and its corresponding triangle in the morphed image—computed an affine transformation for each of the 149 pairs of triangles. This step replicates the procedure for B and the morphed image.

c. Warp triangles: each pixel inside the triangles in A, transferring to a morphed image using affine transform. To generate a warped image of A, repeat the process for each triangle in A—the same way for image B.

4. Blending

In the previous stage were created two warped images for A and B. Using equation (4) for the alpha blending; these two images can be combined.

3.2 Generation Morph Images Using Deep Learning

Combining two or more images in such a way as to provide a seamless transition from one to the next is required the generation of morph images using deep learning. This can be accomplished by utilizing various deep learning strategies, such as Variational Autoencoders (VAEs), Generative Adversarial Networks (GANs), and StyleGAN.

3.2.1 Generative Adversarial Networks

GAN uses two neural networks—a generator and a discriminator. The generator generates new data, whereas the discriminator evaluates it. The training of these two networks is based on the minimax game principle, which implies a two-player presence. The generator attempts to produce more realistic images to deceive the discriminator. In exchange, the discriminator tries to determine whether these images are genuine or fake.

Algorithm 1 explains the mechanism of GAN for the generation of new images [11, 12].

Algorithm (1): Generative Adversarial Network for Face Morph.

Dataset (faces images (

Learned parameters θ and $\emptyset$, morph image.

Begin:

Building a discriminator network ($D_\emptyset$), which consists of several layers of convolutional neural networks and ends with fully connected deep neural networks.

- *Convolution (num_filter, size_filter, stride, padding).*
- *ReLU Function.*
- *Max pooling.*
- *Fully connected.*
- *Sigmoid Function.*

Building a generator network (G_θ), which consists of several layers of a transposed convolutional, ending with the formation of an image with dimensions equal to the dimensions of the input data.

- *UpSampling (Conv2d).*
- *BatchNorm2d.*
- *ReLU.*

Stage 3: *Initialize G_θ and $D_\emptyset$ with random weight θ and $\emptyset$.*

Stage 4: *For number of training iterations **do***

For K with many steps do

Generate m noise samples $(z_1, z_2, z_3, \dots, z_m)$ using G_θ.

Compute loss function for generator:

$$\nabla_{G\,\theta} = \frac{1}{m1}\sum_{i=1}^{m1} \left[log\, log \left(1 - P\left(G(z_i)\right)\right) \right]$$

End loop

For K with many steps do (sample minimum batch of data)

Take m noise samples $(z_1, z_2, z_3, \dots, z_{m1})$ using G_θ for training $D_\emptyset$.

Select m real samples $(x_1, x_2, x_3, \dots, x_{m1})$ from the dataset.

Pre-training $D_\emptyset$ To real and generated samples via minimizing cross entropy.

Compute loss function for discriminator:

$$\nabla_{D\,\emptyset} = \frac{1}{m1}\sum_{i=1}^{m1} \left[log\, log\, P(x_i) + log\, log \left(1 - P\left(G(z_i)\right)\right) \right]$$

End loop

Update network parameters by backpropagation (θ and $\emptyset$)

d loop

Save of the model.

The image of the first subject is entered into the model and generated as a random vector.

The image of the second subject is entered into the model and generated as a random vector.

Combination between vectors to produce a morphed image.

End algorithm

To produce face morphing using GAN, a model must first be trained to understand the structure of faces, and then new images must be generated by combining a pair of faces.

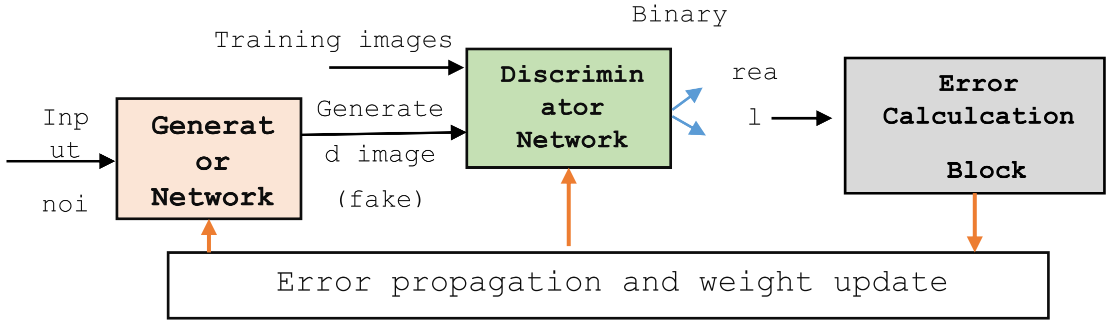


Figure 5: GAN learning to generate and distinguish images.

GAN for face morphing involves the following steps:

1. Gathering Data: Collect vast data for face images.
2. Pre-processing: Applied aligned of the dataset and cropped for input data consistency.
3. Setting up the GAN: Construct a GAN model consisting of two networks: a generator and a discriminator. The generator receives two images as input and generates a morphed image. Then, the output image (morph) quality is evaluated using the discriminator and offers feedback to the generator to enhance its output.
4. Training the GAN: Use the pre-processed data to train the GAN model.
5. Generation morphed: To create faces, enter two input faces for the generator. This will produce faces that are intermediate to the two input faces.
6. Postprocessing: Rescale and save the generated images as a video or sequence of images.

It is essential to consider that the effectiveness of the generated face morphing is reliant on the quality of the data and the efficiency of the model. The model's ability to create realistic morphed faces will improve when the dataset has more variety and represents humans. In addition, adjusting the hyperparameters and training for an adequate number of epochs are required to accomplish the desired level of success.

3.2.2 Variational Autoencoders

Autoencoders are neural networks with extensive outer layers and small inner layers. The beginning and terminating layers match the actual image size. After that, an image is sent via the network, and another image is produced. Backpropagation is used to teach the network after a loss function evaluates its performance. A well-trained network will reproduce the input image. The network's middle layer is low-dimensional [13].

Autoencoders consist of two networks: encoder and decoder. The fundamental concept for the encoder is to convert an input image into a low-dimensional representing latent space. They were followed by a decoder that creates new images from features within latent space. Selecting features randomly from the latent space can produce various images, each with a distinct combination of characteristics.

A variant of that, known as VAEs, seeks to determine whether or not the latent space follows a normal distribution. Because this distribution is normal, any latent vector that falls within that distribution will have a decoded image that is more predictable. A VAEs can be used to produce new data either by randomly selecting new latent vectors or by modifying the values that are already present in the latent vectors [14, 15].

Using VAEs to create a morphed face by entering two face images and converting them into latent space by the encoder. Using a linear combination, we take some features from latent space for the first and another from the second image. It was then decoding these features to generate an output image. This will result in the production of a morphed face that is a mix of the two faces.

VAEs for face morphing involve the following steps:

1. Collect a dataset of face images that will be used to train the variational autoencoder.
2. Pre-process the face images by resizing, normalizing, and augmenting the data if needed.
3. Define the architecture of the VAE, including the encoder and decoder networks. The encoder network compresses the face image into a lower-dimensional latent space while the decoder network reconstructs the original image from the latent space.

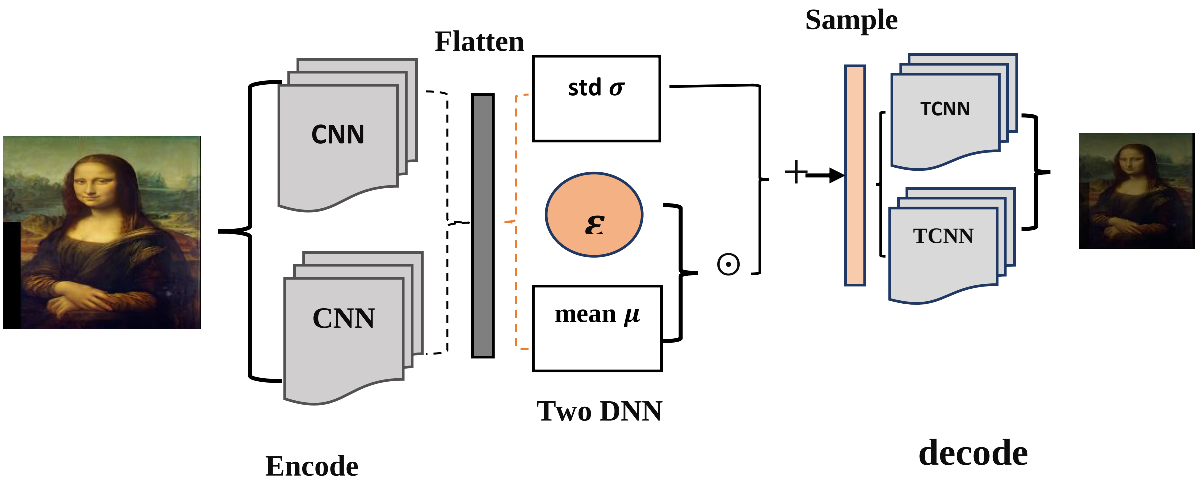


Figure 6: Structure of Variational Autoencoders.

4. Train the VAE using the collected face image dataset. The training process involves minimizing the reconstruction loss and the Kullback-Leibler divergence, which encourages the learned latent space to be smooth and continuous.
5. Once the VAE is trained, generate face morphs between two input faces. To do this, encode both inputs faces into the latent space and then interpolate between their latent representations to generate a smooth transition from one face to another.
6. To obtain the morphed face images, decode the interpolated latent representations using the decoder network.
7. Postprocess the morphed face images by resizing and converting them to the appropriate format, such as JPEG or PNG.

3.3.3 Style Generative Adversarial Networks

StyleGAN is a powerful generative model that has been widely adopted in the field of computer vision and generative art. Which makes it a valuable tool for artists and designers. StyleGAN is a model for deep learning that can create highly realistic human faces with precise control over attributes like facial expression, gender, and age. The model comprises two primary components: a generator that produces new images and a discriminator that evaluates and distinguishes between images in the dataset and fake images. The generator and discriminator work together during training, with the generator attempting to create images that deceive the discriminator while the discriminator aims to identify the fake images accurately [16].

StyleGAN generates high-quality, diversified synthetic images using a complex architecture. StyleGAN is a type of GAN that can produce highly realistic images of faces. Face morphing involves combining features from two or more faces to create a single image that blends the landmarks of each face.

To use StyleGAN for face morphing, first, we needed to train the network on a dataset of faces. Once the network has learned to generate realistic-looking faces, select two faces we want to blend, and the faces should be somewhat similar to create a natural-looking final result.

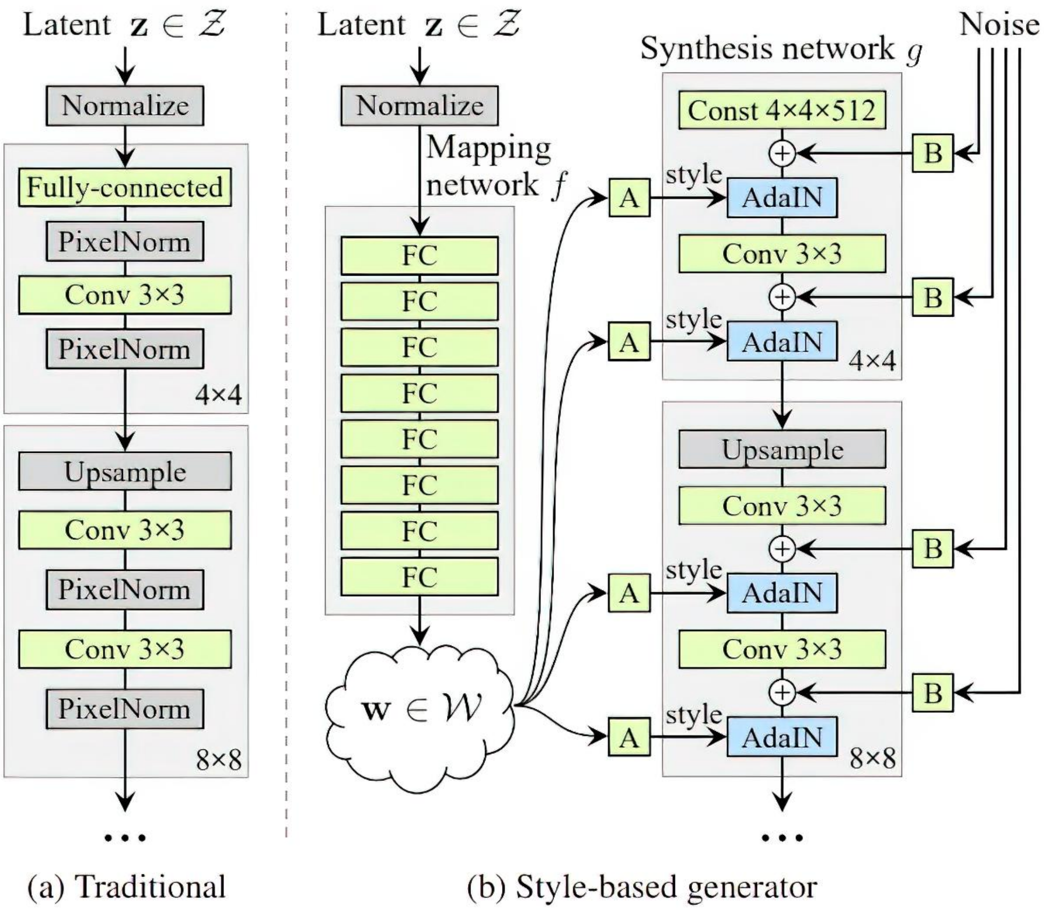


Figure 7: Structural difference between GAN and StyleGAN.

To use StyleGAN for face morphing, the first step is to train the network on a large dataset of faces to generate realistic-looking faces. After that, select two image faces for persons A and B, ensuring they are somewhat similar. The next step is to create separate images of each face using the StyleGAN network by inputting random noise vectors. Once the two images are combined, use image processing techniques to align them using facial landmarks and blend them using alpha blending or morphing techniques. Finally, you would adjust the blending parameters until you achieve a satisfactory result [17, 18, 19, 20].

Algorithm (2): StyleGAN Algorithm.
Dataset (faces images).
Trained network G_θ and $D_\emptyset$.
Building a discriminator network ($D_\emptyset$), which consists of several layers of convolutional neural networks and ends with fully connected deep neural networks. ● *Convolution (num_filter, size_filter, stride, padding).* ● *ReLU Function.* ● *Max pooling.* ● *Fully connected.* ● *Sigmoid Function.*
: Building a generator network ($G_{1\theta}, G_{2\theta}, G_{3\theta}, \dots, G_{n\theta}$), which consists of several layers of a transposed convolutional, ending with the formation of an image with dimensions equal to the dimensions of the input data. ● *Upsampling.* ● *Convolution layer.* ● *Adaptive Instance Normalization (AdaIN).* ● *Style vectors (A) and noise vectors (B).*
Stage 3: *Initialize G_θ and $D_\emptyset$ with random weight θ and $\emptyset$.*

Stage 4: *For number of training iterations* ***do***

For K with many steps do

Generate m noise samples $(z_1, z_2, z_3, \dots, z_m)$ *using* $G_{1\theta}$.

Pass result from $G_{1\theta}$ *to* $G_{2\theta}$ *and* $G_{2\theta}$ *to* $G_{3\theta}$ *until reach to* $G_{n\theta}$.

Compute loss function for generator:

$$\nabla_{G\,\theta} = \frac{1}{m1}\sum_{i=1}^{m1} \left[log\ log\ \left(1 - P\left(G(z_i)\right)\right) \right]$$

End loop

For K with many steps do (sample minimum batch of data)

Take m noise samples $(z_1, z_2, z_3, \dots, z_m)$ *using* G_θ *for training* $D_\emptyset$.

Select m real samples $(x_1, x_2, x_3, \dots, x_m)$ *from the dataset.*

Pre-training $D_\emptyset$ *To real and generated samples via minimizing cross entropy.*

Compute loss function for discriminator:

$$\nabla_{D\,\emptyset} = \frac{1}{m1}\sum_{i=1}^{m1} \left[log\ log\ P(x_i)\ + log\ log\ \left(1 - P\left(G(z_i)\right)\right) \right]$$

End loop

Update network parameters by backpropagation (θ *and* $\emptyset$*)*

nd loop

of the model.

gorithm

Algorithm (3): StyleGAN of Generation Face Morph.

Input: *Two images of real faces for source image and target image.*

Output: *Face morphed.*

Create a well-trained StyleGAN model and recall it.

Determine the number of frames per second and specify the total video frames resulting from a morphing process.

Read an image representing the source.

Read an image representing the target.

Apply pre-processing (change size) for source and target images with the same size.

Enter the source image into the StyleGAN model to convert the latent vector.

Enter the target image into the StyleGAN model to convert the latent vector.

Combine the source image's latent vector with the target image's latent vector using linear interpolating.

Pass latent vector to reconstruction network.

Create a video consisting of a sequence of frames representing stages of transformation from the source image to the target image.

gorithm

4. EXPERIMENTS AND RESULTS

Face morphing is a fascinating computer vision technique that involves blending and transforming the facial features of two or more images to create a new, hybrid image. With advancements in deep learning and computer vision algorithms, it has become possible to generate highly realistic and seamless morphs almost indistinguishable from real images.

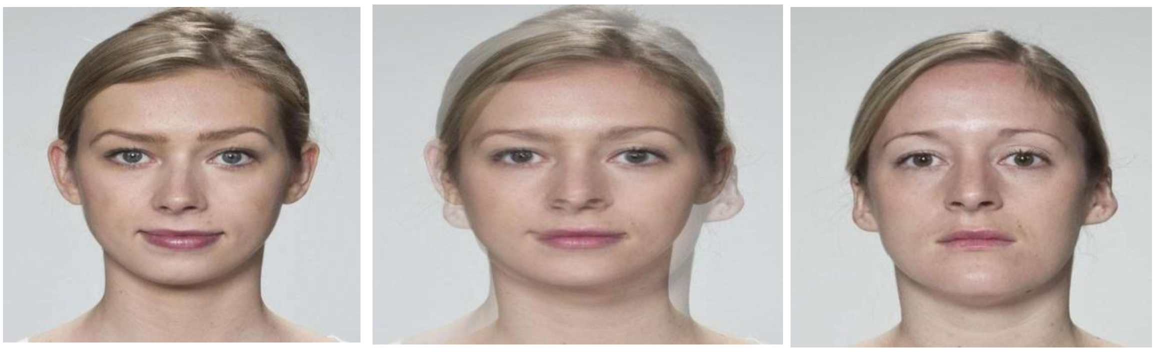
In this discussion, we will explore the results of generated face morphing and examine the benefits and challenges of these techniques.

A convincing visual appearance is essential for morphed images, particularly in scenarios where manual identity verification, such as border checks, or applying for an identity document with a printed photo, without automatic attack detection, is necessary. Where humans are more sensitive to artifacts and distortion in morphed images than software, many programs may be unable to differentiate between them. Our experiments yielded the following key findings and observations.

- The results of our experiments show that generating face morphs using manual landmark-based techniques poses a significant threat to FRS, In the event that the points were determined very accurately on each feature inside the face, starting from the beginnings of the hair and reaching the center of the face. This can be explained by the fact that landmark-based morph generation preserves both the texture and geometric structure of the original images that contribute to the morph. The combination of the two images is very attractive, but it may result in anomalies or distortions that require manual intervention using adobe photoshop or GIMP and image processing techniques to remove. It also requires the content of two images with very similar in features and a snapshot taken. When the difference is considerable or moderate, this method becomes ineffective. These artifacts can be detected by sight but have a high probability of delusional FRS and are acceptable to the system.

Figure 8. Generate morph attack (right and left images authentic images, center: morphed image similar to left and right images).

● The ready-made programs used to generate morph images, as described in the previous section, produce highly distorted images that cannot be used to create morphed face with the perfect appearance. In addition, the majority of these programs specify the images to be transformed within their own database. Others are fine but the image is manually processed by some of the tools available in these programs to remove artifacts.



(a) **(b)** **(c)**

Figure 9: (a) source image (c) target image (b) face morphing using ready-made software.

Another technique for generating a morphed face is an Open CV. Using a Python library ($dlib$), Open CV identifies 68 landmarks automatically. They occupy fixed locations for each face, which may increase or diminish by a certain quantity. The outcomes of this method are depicted in Figure (10).

Figure 10: Samples face morph using Open CV.

Even when there is a high degree of similarity between two images, it is evident from the preceding results that this method is ineffective for creating morph images. The reason for this is that landmarks are automatically determined by the dlib library, where the locations of the landmarks must be aligned in two images. The resultant image is distorted due to variations in locations. There is another reason why this method is not flexible. It only works with JPEG images that are of a suitable size and have very distinct landmarks. The program will not allow images that are small in size.

Because of the space taken by the morphing process and the interest in this field, generation techniques have developed to include deep learning, characterized by its ability to create morphed images of similar and dissimilar faces without distortions.

- VAE is a deep learning technique that is good at generating transformed images, but results in blur around

Figure 11. Generate morph attack (right and left images authentic images, center: morphed image similar to left and right images) using VAE.

- GAN is one of the most effective algorithms for producing realistic-looking fake faces. It is contingent upon either a random vector or a vector derived from the training data. The results are better the longer the training period and the more diverse the data. At the same time, for whatever reason, it results in some distorted images that do not appear to be genuine. The images below illustrate how this technique is used to create a morph compared with landmarks (LM) and StyleGAN. For more information, see.

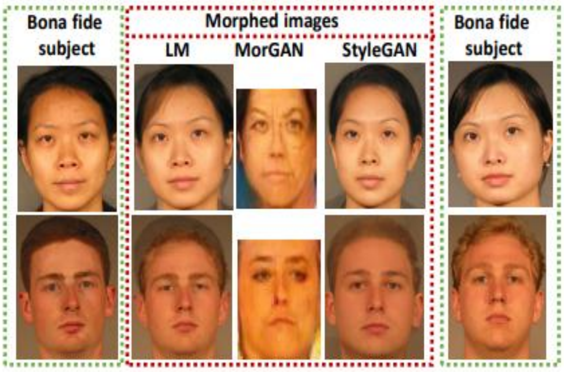


Figure 12. Generate morph attack (right and left images authentic images, center: morphed image similar to left and right images) using GAN.

- StyleGAN is a derivation of the GAN technology. This is one of the best deep learning techniques for creating fake images with a very high degree of accuracy that can induce an optical and electronic illusion. It can create a morphed image between two images without artifacts, blur, or distortion. One of its advantages is that it can generate a morph image from two different images and give another shape that may belong to these two images, while simultaneously being considered a fake image for a different, more realistic person.

StyleGAN has been widely adopted in the field of generative art and has produced some stunning results. However, there are also concerns about the potential misuse of this technology, such as the creation of deep-fakes or other forms of digital manipulation.

Figure 13. Generate morph attack (right and left images authentic images, center: morphed image similar to left and right images) using styleGAN.

StyleGAN provides a wide range of diverse morphed faces. It can generate a spectrum of facial variations encompassing the input faces unique attributes, resulting in distinctive morphs.

The StyleGAN capacity to produce a variety of facial expressions and features ensures a rich set of morphed faces, enhancing the versatility of the morphing process.

We conclude from the above by comparing morphed face-generation techniques. Traditional morphing methods, such as landmark and ready software are relatively simple and intuitive, but they have limitations regarding accuracy and realism, especially when dealing with complex facial expressions and identities. The following method is a deep learning-based approach, which has recently shown promising results in generating highly realistic and diverse morphed faces. These methods, such as VAEs and GAN-based models, rely on large amounts of training data and complex network architectures to learn to generate natural facial features and expressions. While these techniques have achieved impressive results, they can also suffer from limitations such as mode collapse, training instability, and lack of control over the generated output. Recent advances in deep learning-based morphing techniques, including style transfer and attention mechanisms. These approaches aim to address some of the limitations of earlier methods by incorporating additional information, such as facial landmarks or semantic labels, into the generation process. This can result in more fine-grained control over the generated output and improved realism. Overall, while deep learning-based techniques have shown significant promise in generating highly realistic and diverse morphed faces, several challenges still need to be addressed, including the need for large amounts of training data, the issue of ethical considerations, and the potential biases in the generated output. Nevertheless, the continued development of these techniques is likely to lead to exciting new applications.

5. CONCLUSIONS

The generation of morphed faces has become an increasingly important area of research in recent years, with a wide range of applications such as biometric identification, entertainment, and social media. In this comparison of morphed face-generation techniques, we have explored various approaches to this task and evaluated their strengths and weaknesses. They started by discussing traditional morphing methods, such as linear interpolation and mesh warping, which have been widely used in computer graphics for many years. While these techniques are relatively simple and intuitive, they have limitations regarding accuracy and realism, especially when dealing with complex facial expressions and identities.

The next explored deep learning-based approaches, which have recently shown promising results in generating highly realistic and diverse morphed faces. These methods, such as GAN-based models, rely on large amounts of training data and complex network architectures to learn to generate natural facial features and expressions. While these techniques have achieved impressive results, they can also suffer from limitations such as mode collapse, training instability, and lack of control over the generated output.

We finally discussed recent advances in deep learning-based morphing techniques, including style transfer and attention mechanisms. These approaches aim to address some of the limitations of earlier methods by incorporating additional information, such as facial landmarks or semantic labels, into the generation process. This can result in more fine-grained control over the generated output and improved realism. Overall, while deep learning-based techniques have shown significant promise in generating highly realistic and diverse morphed faces, several challenges still need to be addressed, including the need for large amounts of training data, the issue of ethical considerations, and the potential biases in the generated output. Nevertheless, the continued development of these techniques is likely to lead to exciting new applications in fields such as entertainment, advertising, and social media.

REFERENCES

1. Y. Kortli, M. Jridi, A. al Falou, M. Atri, Face recognition systems: A survey, Sensors. 20 (2020) 342. https://doi.org/10.3390/s20020342.
2. R. V. Virgil Petrescu, Face Recognition as a Biometric Application, Mechatronics and Robotics. 3 (2019) 237–257. https://doi.org/ 10.3844/jmrsp.2019.237.257.
3. J. S. del Rio, D. Moctezuma, C. Conde, I. M. de Diego, E. Cabello, Automated border control e-gates and facial recognition systems, computers & security. 62 (2016) 49-72. https://doi.org/ 10.1016/j.cose.2016.07.001.
4. O. O. Petrova, K. B. Bulatov, Methods of machine-readable zone recognition results post-processing, In: Eleventh International Conference on Machine Vision (ICMV). 11041 (2019) 387-393. https://doi.org/10.1117/12.2522792.
5. S. Noori, Suspicious infrastructures: automating border control and the multiplication of mistrust through biometric e-gates, Geopolitics. 27(4) (2022) 1117-1139. https://doi.org/10.1080/14650045.2021.1952183.
6. D. Ortega, A. Fernández-Isabel, I. Martín de Diego, C. Conde, E. Cabello, Dynamic facial presentation attack detection for automated border control systems, Comput Secur. 92 (2020) 101744. https://doi.org/10.1016/J.COSE.2020.101744.
7. A. Makrushin, T. Neubert, J. Dittmann, Automatic generation and detection of visually faultless facial morphs. In: 12th International Joint Conference on Computer Vision Theory and Applications. 7 (2017) 39–50. https://doi.org/10.5220/0006131100390050.
8. U. M. Kelly, L. Spreeuwers, R. Veldhuis, Improving Deep-Learning-based Face Recognition to Increase Robustness against Morphing Attacks. In: CS & IT Conference Proceedings. 10 (2020) 1–12. https://doi.org/10.5121/csit.2020.101901.
9. U. Scherhag, L. Debiasi, C. Rathgeb, C. Busch, A. Uhl, Detection of Face Morphing Attacks Based on PRNU Analysis, Transactions on Biometrics, Behavior, and Identity Science. 1 (2019) 302–317. https://doi.org/10.1109/TBIOM.2019.2942395.
10. S. Venkatesh, R. Ramachandra, K. Raja, C. Busch, Single Image Face Morphing Attack Detection Using Ensemble of Features, In: 23rd International Conference on Information Fusion, (2020) 1-6. https://doi.org/10.23919/FUSION45008.2020.9190629.
11. L. DeBruine, B. Jones, Face Research Lab London Set, (2017). https://figshare.com/articles/dataset/Face_Research_Lab_London_Set/5047666/ (accessed May 30, 2017).
12. J. Bird, Football players and staff faces, dataset for project Generating a Football Team with Progressive GAN (PGAN) and Char-RNN, (2020). https://www.kaggle.com/datasets/birdy654/football-players-and-staff-faces/ (accessed November 27, 2022).
13. L. DeBruine, B. Jones, Young Adult White Faces with Manipulated Versions, (2022). https://figshare.com/articles/dataset/Young_Adult_White_Faces_with_Manipulated_Versions/4220517/ (accessed November 27, 2022).
14. J. M. Chen, J. B. Norman, Y. Nam, Broadening the stimulus set: Introducing the American Multiracial Faces Database, Behav Res. 53 (2021) 371–389. https://doi.org/doi: 10.3758/S13428-020-01447-8.
15. M. Walker, S. Schönborn, R. Greifeneder, T. Vetter, the basel face database: A validated set of photographs reflecting systematic differences in big two and big five personality dimensions, (2018). https://doi.org/10.1371/JOURNAL.PONE.0193190.
16. S. Adil Saribay, A. F. Biten, E. O. Meral, P. Aldan, V. Trebicky, K. Kleisner, The Bogazici face database: Standardized photographs of Turkish faces with supporting materials, (2018). https://doi.org/10.1371/JOURNAL.PONE.0192018.
17. D. S. Ma, J. Kantner, B. Wittenbrink, Chicago Face Database: Multiracial expansion, 53 (2021)1289–1300. https://doi.org/10.3758/S13428-020-01482-5.
18. N. Strohminger, K. Gray, V. Chituc, J. Heffner, C. Schein, C., T. B. Heagins, MR2 face database, (2015). https://doi.org/10.3758/T45216-060-00322-5.
19. S. Venkatesh, H. Zhang, R. Ramachandra, K. Raja, N. Damer, C. Busch, International Workshop on Biometrics and Forensics (IWBF), Transactions on Biometrics, Behavior, and Identity Science, 3 (2020) 154 - 154. https://doi.org/10.1109/IWBF49977.2020.9107970
20. T. Neubert, A. Makrushin, M. Hildebrandt, C. Kraetzer, J. Dittmann, Extended StirTrace benchmarking of biometric and forensic qualities of morphed face images, IET Biometrics, 7 (2018) 325–332.