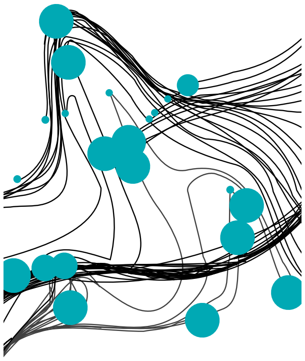
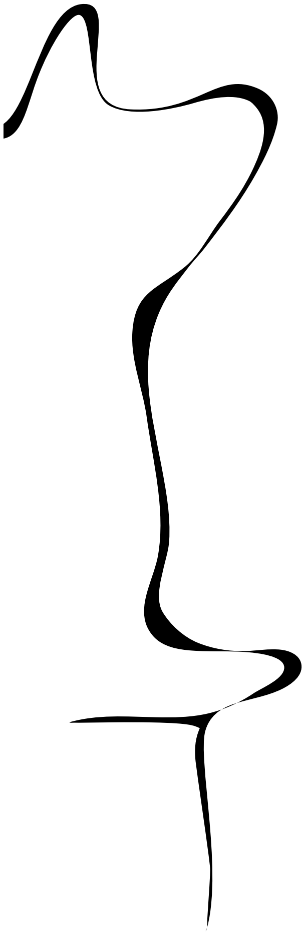


.03413

DMB

DATA MANAGEMENT
AND
BIOMETRICS



STYLEDEMORPHER: HIGH-QUALITY FACE DEMORPHING USING STYLEGAN2'S LATENT SPACE

Raul Ismayilov

MASTER'S ASSIGNMENT

Committee:
dr.ir. L.J. Spreeuwers
I. Batskos MSc
dr.ing. G. Englebienne

August, 2024

Data Management and Biometrics
EEMathCS
University of Twente
P.O. Box 217
7500 AE Enschede
The Netherlands

UNIVERSITY OF TWENTE. | DIGITAL SOCIETY INSTITUTE

Abstract—**Morphing attacks pose a substantial threat to Automated Border Control (ABC) systems by enabling the creation of identity documents used by multiple individuals, thereby compromising border security. Face demorphing has emerged as a crucial technique to counteract these attacks, aiming to disentangle and reconstruct the concealed identities within a morph. This paper introduces a novel face demorphing framework leveraging StyleGAN2's latent space. The framework includes an advanced encoder, ReStyle-ID, designed to embed identities into StyleGAN2's latent space with high accuracy, and StyleDemorpher, a specialized face demorphing network trained on the newly created DemorphDB dataset. DemorphDB features high-quality morph images, providing a challenging and realistic training environment for the StyleDemorpher.**

The ReStyle-ID encoder and StyleDemorpher frameworks collectively enhance the accuracy and quality of face demorphing, addressing the limitations of previous approaches such as low resolution and poor generalizability. The ReStyle-ID encoder utilizes improved loss functions and training data, achieving improvements in identity reconstruction when compared to other encoding methods. StyleDemorpher excels in reconstructing high-quality demorphed images, demonstrating high generalizability across various morphing methods and unseen identities. This work introduces a robust solution for face demorphing and sets the stage for future advancements by developing a comprehensive dataset and a scalable framework for continued research and development in biometric security.

Index Terms—**Face Demorphing, GAN, Deep Learning, Face Recognition**

I. Introduction

Morphing attacks present a significant threat to Automated Border Control (ABC) systems [1], as they enable the creation of identity documents that can be used by multiple individuals whose features are blended in the morph. This vulnerability can potentially allow two identities to share a single document, undermining the integrity of border security measures [2], [3]. First introduced by [4], face demorphing has emerged as a prominent research topic in biometrics due to its potential to counteract morphing attacks. The primary objective of face demorphing is to disentangle the two identities embedded within a morph. This often involves reconstructing the second identity, which is not physically present at the ABC gate but is concealed within the morph.

Face demorphing faces several significant challenges, primarily due to the lack of prior information about the morphing method and the blending factor used to combine the two identities. Additionally, the live image capture at the ABC gate often differs from the one used to generate the morph in terms of illumination, pose, and expression. These factors make the exact reconstruction of the second identity through facial landmarks complex and prone to noticeable artifacts [4].

To overcome these challenges, researchers have increasingly explored deep learning-based approaches for face demorphing. Techniques utilizing Convolutional Neural Networks (CNNs) and Generative Adversarial Networks (GANs) have shown promise [5], [6], [7], [8]. However, these networks often reconstruct low-resolution images with artifacts and distortions. Moreover, their training on limited datasets can lead to poor generalizability when applied to previously unseen morphing methods and identities.

This paper addresses the limitations of current deep learning-based methods for face demorphing by introducing a novel approach leveraging the StyleGAN2's [9] latent space. To overcome the challenge of low-resolution reconstructions, StyleGAN2 is utilized, which is capable of generating images at a resolution of 1024×1024 pixels. A newly developed encoder framework, ReStyle-ID, is designed to accurately embed existing identities into the latent space of StyleGAN2.

The ReStyle-ID encoder network is trained on over 100,000 different identities to ensure its generalizability. Using transfer learning [10], a specialized StyleDemorpher network is further trained specifically for the demorphing task. This training is conducted on a newly created dataset, DemorphDB, introduced in this paper. The morph images in DemorphDB are of high quality, providing a complex and realistic challenge for the StyleDemorpher to learn from.

Overall, the proposed framework simulates realistic morphing attack scenarios and aims to reconstruct accurate and high-quality demorphed images. This method demonstrates high generalizability across unseen identities and morphing methods, effectively addressing the shortcomings of previous approaches.

Thus, the contributions of this paper are threefold:

- **A Novel Demorphing Database:** Introducing DemorphDB, a new demorphing database featuring high-quality, passport-like images of 1653 identities from five public datasets, including 643 identities with multiple images. For each identity, both traditional and deep learning-based high-quality morphs, chosen from the ten closest identities, are provided. The database is designed for extensibility, allowing for additional morphing methods.
- **Improved StyleGAN2 Encoder:** Presenting ReStyle-ID, an enhanced StyleGAN2 [9] encoder network that better retains identity information in input face images. Using an iterative encoder based on the ReStyle [11] architecture, with improved loss functions and training data, this encoder surpasses the original ReStyle in preserving identity information and places encodings more favorably within the latent space. It is also three orders of magnitude faster than optimization-based methods [9], making it significantly more efficient.
- **High-Quality Demorphing Network:** Introducing StyleDemorpher, a demorphing network that produces high-quality and accurate demorphs without needing prior knowledge of the morphing method, relying solely on training examples. The network generalizes well to unseen datasets and morphing techniques, trained on high-quality morphs with minimal artifacts. It utilizes the ReStyle-ID encoder, retrained to use a morph image and a live capture to reconstruct the second identity within the morph, offering a practical solution without requiring access to the original images used for morphing.

II. Related Work

A. Face Demorphing

Face morphing combines features from two distinct identities into a single image that shares attributes of both. This process is defined as follows for images I_A and I_B of two different identities:

$$I_{AB} = \mathcal{M}\left(I_A, I_B\right), \tag{1}$$

where $\mathcal{M}\left(.\right)$ denotes the face morphing technique.

There are primarily two approaches to generating morphs: landmark-based and deep learning-based. Landmark-based methods often involve using facial landmarks to create triangular meshes. These meshes are then warped to produce a morph [12]. These methods often lead to ghosting artifacts that typically require manual retouching to achieve convincing results. Various landmark-based morphing techniques have been proposed in literature [13], [14], [15], with most automatic methods employing a splicing technique to integrate the morphed facial region seamlessly into one of the original identity images [16].

Deep learning-based methods [17], [18], [19], in contrast, eliminate the need for landmarks by leveraging neural network architectures for end-to-end morph generation. While these methods generally reduce the need for manual adjustments, they can sometimes yield lower quality results [20].

Face demorphing, first introduced in [4] and building on prior research [2], [21], aims to reconstruct the image of an accomplice, I_B, from a forged document featuring a morphed image I_{AB}. This task becomes challenging when the criminal identity A attempts to use the document, particularly because the exact method of morph generation is often unknown. Complications are further amplified since the image of identity A captured at the ABC gate differs from the image used to generate the morph, introducing potential artifacts during the demorphing process as highlighted in [4]. To address one of these challenges, [22] explores using a deep learning network to first estimate the morphing factor before performing face demorphing. However, both of these methods apply only to conventional landmark-based morphs and rely on the minimal error of landmark detection algorithms.

For this reason, deep learning-based face demorphing methods could address this challenge by training on morphs generated by various morphing methods. Several deep learning-based face demorphing methods have been proposed in literature. For instance, [5] describes a Convolutional Neural Network (CNN) that processes both the document and live capture images to output the demorphed image. Another approach, FD-GAN [7], employs a Generative Adversarial Network (GAN) framework consisting of an encoder network, an identity separation network for disentangling the encoded images to recover features of the accomplice, and a restoration network to reconstruct the image based on these features. The discriminator network then evaluates the authenticity of the generated image compared to the target image. MorphGANFormer [8] utilizes the GANformer [23] architecture, embedding real images into the latent space using an optimization method driven by identity-related loss functions. Following the embedding, a simple linear interpolation of the latent codes is used to generate the accomplice's identity.

While landmark-based demorphing methods are effective only under specific conditions, when a landmark-based morphing method is used and landmarks are precisely extracted, they often generate artifacts and are ill-suited for handling attacks using deep learning-based morphing techniques. Conversely, while deep learning-based demorphing methods could be trained to handle various morph types, they often suffer from limitations such as training on a limited number of identities, using low-resolution images, and producing distorted images due to insufficient training data. An exception is MorphGANFormer [8], which uses a pre-trained GANformer [23] capable of generating high-quality, high-resolution facial images. However, its effectiveness is limited and currently only proven on GANformer-based morphs, suggesting that its simple interpolation method may not be effective with other morphing techniques due to potential lack of disentanglement of features in the latent space with respect to the morphing method.

B. Latent Space of StyleGAN2

Introduced in [9], StyleGAN2 is an advanced iteration of the StyleGAN network [24]. StyleGAN employs a novel GAN generator architecture capable of producing high-resolution images. Unlike traditional generator networks that start from a latent code [25], [26], StyleGAN initiates from a constant input and incorporates one or multiple different latent codes (also referred to as "styles") at the input of each convolutional layer. Furthermore, each layer is enhanced with independently scaled noise inputs to introduce fine-grained stochastic details. This architecture facilitates the generation of high-resolution, realistic images that can be finely controlled via the styles. Overall, the StyleGAN architecture consists of 18 convolutional blocks, each receiving latent codes corresponding to individual styles. Moreover, StyleGAN features a mapping network that transforms the latent codes, sampled from a multivariate standard normal distribution $\mathcal{Z}$, into an intermediate latent space $\mathcal{W}$. This transformation achieves a higher disentanglement of the latent variables associated with various image attributes. StyleGAN2 [9] refines this generator architecture to eliminate artifacts observed in some generated images and to enhance training stability.

Given the high quality and disentanglement of the StyleGAN2 latent space, numerous studies have explored mapping real images into this space for editing purposes. For instance, starting with a neutral facial expression, StyleGAN2 can be used to identify and modify the corresponding latent code to reflect a smiling expression in the resulting image. This process begins with the embedding of the real image into the StyleGAN2 latent space, a technique known as GAN inversion [27]. Two primary methods are often employed for this embedding: optimization-based and encoder-based.

Optimization-based methods, as discussed in [9], [28], [29], iteratively refine the latent codes to minimize the disparity between the target and generated images, using gradient descent combined with various loss functions. Although this method yields high resemblance, it is computationally intensive, often requiring several minutes to embed a single image. Conversely, encoder-based methods, such as those found in [30], [31], and [11], utilize trained encoder networks to map images into the latent space more quickly, typically within a single or a few forward passes, despite generally achieving lower fidelity compared to optimization-based methods.

Following embedding, the attributes of the image can be altered using the StyleGAN2 latent space. For facial images, several studies [32], [33], [34] have demonstrated the ability to modify features such as expression, pose, or illumination by editing the latent codes. However, these methods are not directly applicable to the face demorphing task, which requires identifying a novel facial identity within the latent space that corresponds to an identity hidden in a morphed image. Therefore, although embedding is still necessary for face demorphing to project real faces into the StyleGAN2 latent space, this paper introduces a novel approach tailored to the face demorphing task.

III. Methodology

In this section, the methodology behind the proposed frameworks is detailed. The core component, the ReStyle-ID encoder framework, is first introduced, highlighting its role in identity-preserving inversion necessary for the face demorphing task in the StyleGAN2 [9] latent space. Next, the StyleDemorpher face demorphing framework is presented, demonstrating how it modifies the pre-trained ReStyle-ID encoder to achieve high-quality face demorphing results. Finally, the formulation of the loss functions for both frameworks is discussed.

A. ReStyle-ID: Identity-Preserving Inversion Framework

The architecture and operation of the proposed ReStyle-ID framework are largely similar to ReStyle [11]. Unlike conventional StyleGAN2 encoders such as e4e [31] or pSp [30], which encode the input image in a single pass, ReStyle uses several iterative forward passes, each improving the encoding. The proposed ReStyle-ID framework enhances the ReStyle encoder specifically for identity information preservation, crucial for face demorphing where the morphed image is highly similar to the identity to be recovered. These improvements were achieved through the following modifications:

- Utilization of a larger dataset and the inclusion of synthetic images of passport-like quality, further described in Section V-A1.
- Enhancement of the identity loss by using the MTCNN [35] model for face detection and cropping instead of a fixed center crop, and adding the MS-SSIM [36] loss function, further described in Section III-C3.
- Complete removal of background information in the images to be encoded, ignoring out-of-domain background information that can lead encodings outside the well-defined regions of the StyleGAN2 latent space.

The operation of the ReStyle-ID framework is illustrated in Figure 1. Given an input image I_x, the objective of the ReStyle-ID framework is to find a latent code $\mathbf{w}$ that best represents the input. The expanded latent space of StyleGAN2, denoted as $\mathcal{W}+$, is utilized for this task. Unlike the conventional latent space $\mathcal{W}$ of StyleGAN2, which uses a single 512-dimensional latent code (style) $\mathbf{w}$ for all 18 layers, the $\mathcal{W}+$ space allows for 18 different $\mathbf{w}$ vectors, significantly improving the inversion quality [28].

Initially, the latent code $\mathbf{w}_{\hat{y}_0}$ is set to the average latent code of StyleGAN2, $\overline{\mathbf{w}}$, with its corresponding image $I_{\hat{y}_0}$. At each iteration t, where $0 \leq t \leq N$ and N is the total number of iteration steps, the target image I_x and the current prediction $I_{\hat{y}_t}$ are concatenated and passed to the encoder network E. The architecture of the encoder network is visualized in Figure 2. This network generates a residual code $\mathbf{\Delta}_t^E$:

$$\mathbf{\Delta}_t^E = E\left(I_x \parallel I_{\hat{y}_t}\right). \tag{2}$$

The residual code is then combined with the current latent code prediction $\mathbf{w}_{\hat{y}_t}$, resulting in an improved latent code:

$$\mathbf{w}_{\hat{y}_{t+1}} = \mathbf{\Delta}_t^E + \mathbf{w}_{\hat{y}_t}. \tag{3}$$

The StyleGAN2 generator G then generates the image $I_{\hat{y}_{t+1}}$ corresponding to the improved latent code:

$$I_{\hat{y}_{t+1}} = G\left(\mathbf{w}_{\hat{y}_{t+1}}\right). \tag{4}$$

This process continues iteratively, updating current latent code and corresponding image until the final iteration N.

During training, an additional step is performed using a pre-trained face segmentation network [37] to identify and remove background pixels in both I_x and $I_{\hat{y}_{t+1}}$, setting their values to zero. This effectively eliminates the background from the images, allowing the encoder network to focus solely on the identity within the image. After this operation, the loss function described in Section III-C6 is calculated, and the back-propagation algorithm is applied. It is important to note that the StyleGAN2 generator G remains frozen, and only the weights of the encoder network E are updated.

B. StyleDemorpher: Face Demorphing Framework

The StyleDemorpher framework excels in face demorphing by leveraging the latent space capabilities of StyleGAN2 [9]. It adopts the ReStyle-ID encoder architecture, as shown in Figure 2. The pre-trained weights of the ReStyle-ID encoder are used as the starting point for training StyleDemorpher. This strategic use of pre-trained weights equips StyleDemorpher with a robust initial understanding of the correlation between image representations and the latent space of StyleGAN2, built from a substantial dataset used with the ReStyle-ID encoder.

The ReStyle-ID encoder's ability to train with single images of varying expressions and poses enables the use of extensive image datasets such as FFHQ [9] and CelebA-HQ [40]. However, modeling face morphing attacks requires high-quality, passport-like images. Additionally, authorities typically only

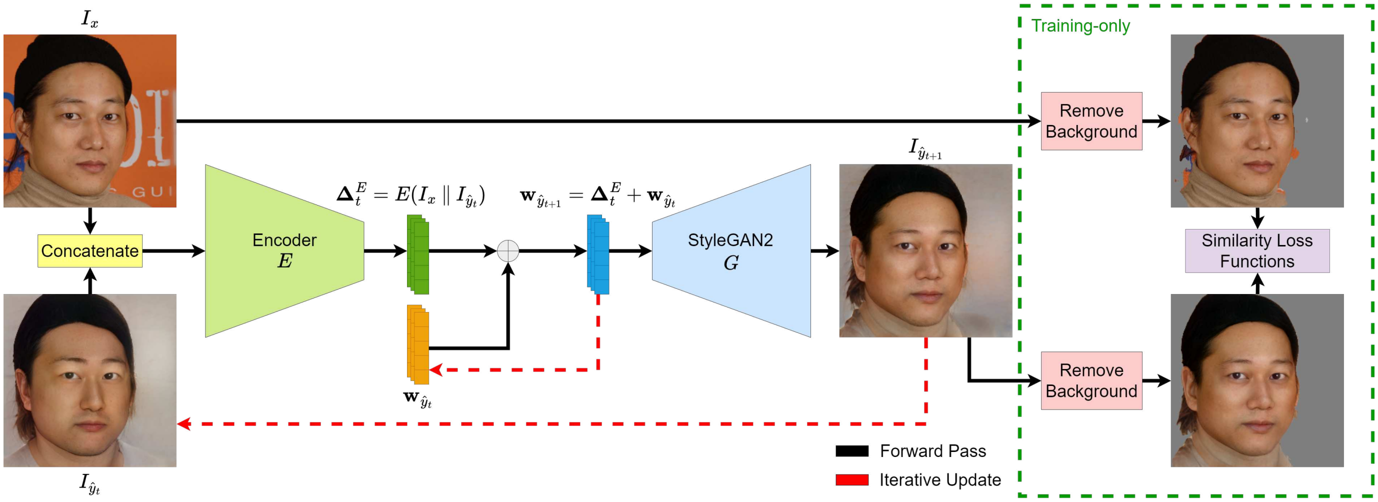


Figure 1: **ReStyle-ID inversion framework.** This framework builds upon ReStyle [11] with enhancements to better encode identity information during the training process. The input image, I_x, begins with $\mathbf{w}_{\hat{y}_0}$ and $I_{\hat{y}_0}$ initialized to the mean StyleGAN2 latent code and its corresponding image, respectively. At each iteration step t, I_x and $I_{\hat{y}_t}$ are concatenated along the channel dimension and fed into the encoder network E. The encoder's task is to find the residual code $\mathbf{\Delta}_t^E$, which is added to the current latent code $\mathbf{w}_{\hat{y}_t}$. This adjustment aims to produce a new latent code $\mathbf{w}_{\hat{y}_{t+1}}$ that more closely resembles I_x when forwarded through StyleGAN2. These updates are iteratively refined at each step. Note that all latent codes $\mathbf{w} \in \mathcal{W}+$. During training, a segmentation model removes the backgrounds of I_x and the generated $I_{\hat{y}_{t+1}}$ to compute the similarity-based loss functions, ensuring the inversion process focuses on the subject rather than the background.

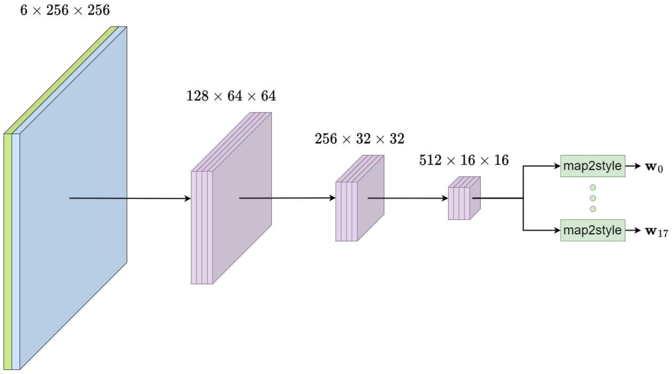

$6 \times 256 \times 256$
$128 \times 64 \times 64$
$256 \times 32 \times 32$
$512 \times 16 \times 16$
map2style
$\mathbf{w}_0$
map2style
$\mathbf{w}_{17}$

Figure 2: Simplified architecture of the ReStyle-ID encoder and StyleDemorpher network, following [11]. The two input images are concatenated along the channel dimension, and feature maps are extracted using the feature pyramid network [38] based on the ResNet-IR [39] backbone. The feature maps are passed through 18 *map2style* networks [30], transforming them into 18 512-dimensional vectors corresponding to $\mathbf{w} \in \mathcal{W}+$.

have access to the morphed images in documents and live captures of individuals using these documents, not the original images used to create the morphs. To simulate this scenario for training StyleDemorpher, a dataset must include at least two distinct images of the same individual - one to generate the morph and another representing the person's live capture at the ABC gate. This requirement limits the data available for training the face demorphing network. By initializing StyleDemorpher with weights from the encoder network, which already establishes a connection between image and latent space, overfitting on the smaller dataset can be mitigated. This strategy enhances the generalizability to unseen identities and various morphing methods.

Before detailing the StyleDemorpher framework's design, it is essential to define several terms related to the dataset used in training. This dataset comprises quadruplets of images, denoted as $(I_A, I_{A'}, I_B, I_{AB})$ and further described in Section IV. The definitions of these images are as follows:

- I_A - An image of a criminal, A, used to create the morph.
- $I_{A'}$ - A different image of the same criminal, A, modeled as the live capture at the ABC gate.
- I_B - An image of an accomplice, B, assisting criminal A in the morph creation. This image is targeted for recovery by the face demorphing algorithm.
- I_{AB} - The morph image, used in the identity document that criminal A attempts to utilize.

Although all four images can be employed during the StyleDemorpher's training phase, only I_{AB} and $I_{A'}$ are available during inference, as the images used to create the morph are not accessible.

The operation of the StyleDemorpher framework is depicted in Figure 3. The framework processes an input morph image, I_{AB}, and a live capture image, $I_{A'}$. While Figure 3 visualizes the morph image generated using StyleGAN2, any morphing method can be used for generating morph images. I_{AB} and $I_{A'}$ are concatenated along the channel dimension and fed into

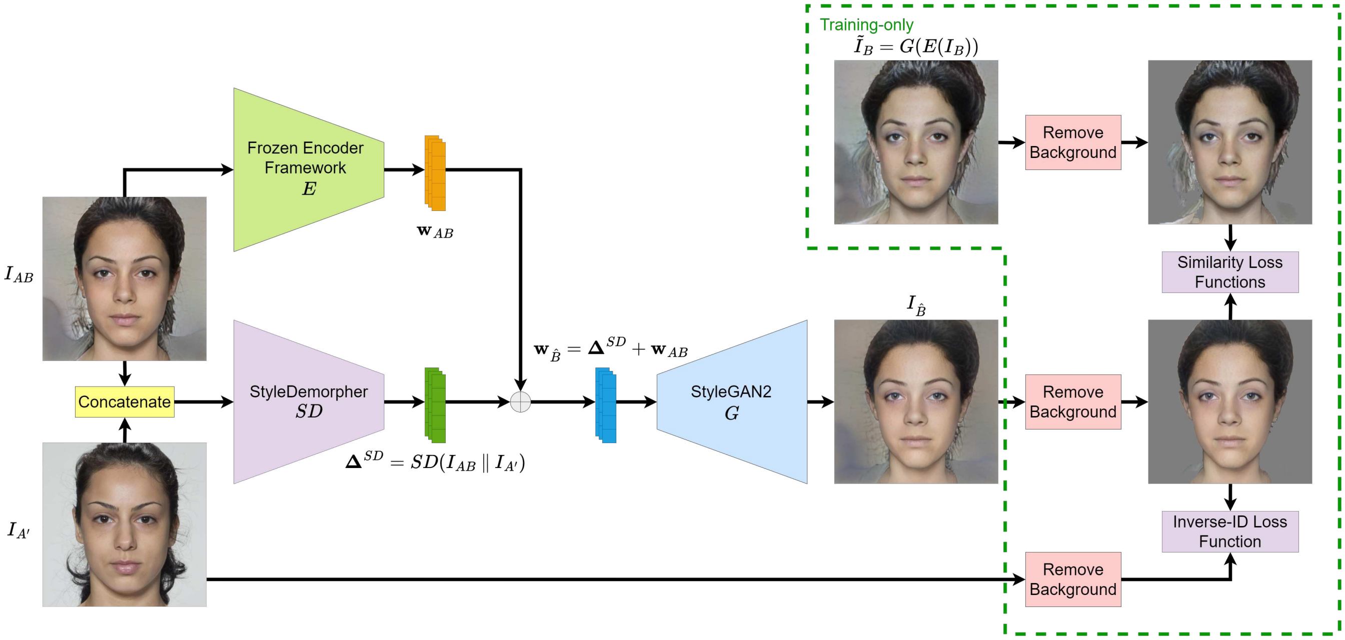

Training-only
$\tilde{I}_B = G(E(I_B))$
Remove Background
Similarity Loss Functions
Frozen Encoder Framework E
$\mathbf{w}_{AB}$
I_{AB}
$I_{\hat{B}}$
$\mathbf{w}_{\hat{B}} = \mathbf{\Delta}^{SD} + \mathbf{w}_{AB}$
Concatenate
StyleDemorpher SD
StyleGAN2 G
Remove Background
$\mathbf{\Delta}^{SD} = SD(I_{AB} \parallel I_{A'})$
Inverse-ID Loss Function
$I_{A'}$
Remove Background

Figure 3: **StyleDemorpher face demorphing framework.** This framework utilizes the transfer learning mechanism [10] by initializing the weights of the StyleDemorpher network, which has the same architecture as the ReStyle-ID encoder, with the weights of ReStyle-ID. The ReStyle-ID encoder framework is also used with frozen weights to encode the input image I_{AB} into the latent code $\mathbf{w}_{AB}$, corresponding to the latent code generated in the final iteration of ReStyle-ID. In the forward pass procedure, the morph image I_{AB} and the live capture image $I_{A'}$ are concatenated and used as input for the StyleDemorpher network, SD. The network calculates the residual code $\mathbf{\Delta}^{SD}$, which, when added to $\mathbf{w}_{AB}$, generates the latent code $\mathbf{w}_{\hat{B}}$ that aims to recover the identity of B present within the morph AB. StyleGAN2 is then used to generate the image $I_{\hat{B}}$ from $\mathbf{w}_{\hat{B}}$. During the training, the target image I_B is first inverted into the StyleGAN2 latent space using ReStyle-ID framework, i.e., $\tilde{I}_B = G(E(I_B))$. Following this, the backgrounds of $I_{\hat{B}}$, $\tilde{I}_B$ and $I_{A'}$ are removed, and the similarity-based loss functions are computed between $I_{\hat{B}}$ and $\tilde{I}_B$, while inverse identity loss is computed between $I_{\hat{B}}$ and $I_{A'}$.

the StyleDemorpher network, denoted as SD. This network outputs a residual code, $\mathbf{\Delta}^{SD}$:

$$\mathbf{\Delta}^{SD} = SD\left(I_{AB} \parallel I_{A'}\right). \tag{5}$$

Concurrently, the morph image I_{AB} is input to the frozen, pre-trained ReStyle-ID framework. Although simplified in the figure, the "Frozen Encoder Framework" block represents the complete encoding framework shown in Figure 1. The encoding is iteratively refined over N steps, and the final latent code $\mathbf{w}_{AB}$ at $t = N$ is saved. This latent code is then combined with $\mathbf{\Delta}^{SD}$ to estimate the latent code for reconstructing identity B, $\mathbf{w}_{\hat{B}}$:

$$\mathbf{w}_{\hat{B}} = \mathbf{\Delta}^{SD} + \mathbf{w}_{AB}. \tag{6}$$

This approach leverages the morph's latent code to navigate to the latent space location of identity B. Using this latent code, the frozen StyleGAN2 generator, G, reconstructs the image of identity B:

$$I_{\hat{B}} = G\left(\mathbf{w}_{\hat{B}}\right). \tag{7}$$

During training, the target image of identity B guides the StyleDemorpher's learning process. Instead of using I_B directly, ReStyle-ID encodes it into the latent space, and StyleGAN2 reconstructs it as $\tilde{I}_B = G(E(I_B))$, promoting demorphing within StyleGAN2's latent space. Following this, the backgrounds of $I_{\hat{B}}$, $\tilde{I}_B$, and $I_{A'}$ are removed, and similarity-based loss functions are computed between $\tilde{I}_B$ and $I_{\hat{B}}$. Additionally, inverse identity loss, aimed at removing the presence of identity $I_{A'}$ in $I_{\hat{B}}$, is computed. The loss computation is further discussed in Section III-C7.

During the initial development stages, the demorphing process fully conducted within the latent space of StyleGAN2 was explored. In this approach, the inputs for StyleDemorpher network were latent codes instead of images. While this method proved effective for editing attributes in StyleGAN2, such as changing a neutral expression to a smile, as demonstrated in previous work [33], it resulted in only minimal changes when applied to face demorphing. A detailed explanation of this behavior is provided in Appendix A.

C. Loss Function Formulation

This section introduces the individual loss functions used in training the ReStyle-ID and StyleDemorpher frameworks. Since most of the individual loss functions are utilized by both frameworks, a simpler notation is adopted using variables x

and y, representing two different images used in the computation of a specific loss. The final training objectives for both frameworks are then presented, utilizing framework-specific notations.

*1) **L2 loss**:* Pixel-wise L2 Loss, also known as Mean Squared Error (MSE) loss, is a fundamental and widely-used loss function when it comes to training deep learning models. It is defined as follows:

$$\mathcal{L}_{\text{L2}}\left(x,y\right)=\left\|x-y\right\|_{2}. \tag{8}$$

*2) **Perceptual loss**:* Perceptual loss is widely used in training Convolutional Neural Networks (CNNs) and Generative Adversarial Networks (GANs) [41], [42], [43] as it helps these networks learn perceptual similarities between images. In this work, LPIPS [44] loss, based on the AlexNet [45] backbone, is utilized over the standard perceptual loss [46]. Early experiments and previous research [47] have shown that LPIPS loss better preserves image quality and sharpness. The LPIPS loss is defined as:

$$\mathcal{L}_{\text{LPIPS}}\left(x,y\right)=\left\|F\left(x\right)-F\left(y\right)\right\|_{2}, \tag{9}$$

where F represents the AlexNet perceptual feature extraction network.

*3) **Identity loss**:* Identity loss is crucial during the training of both ReStyle-ID and StyleDemorpher frameworks. This loss helps preserve identity-related features within the image, which is essential for the face demorphing procedure. The identity loss is defined as:

$$\mathcal{L}_{\text{ID}}\left(x,y\right)=1-S_{c}\left(R\big(M(x)\big),R\big(M(y)\big)\right), \tag{10}$$

where S_c represents the cosine similarity metric, R is the pre-trained ArcFace [39] network specialized in facial recognition and verification, and M is the pre-trained MTCNN [35] network used for automatic face detection and cropping.

An improvement introduced in this work, compared to [30] and [11], is the use of automatic face detection. Instead of performing a simple center crop of the face image before passing it to the ArcFace network, the MTCNN network detects the bounding box around the face. The cropping and resizing are then performed, and the resulting image is passed to the ArcFace network. This makes the loss implementation more robust, allowing it to handle images with varying poses or facial structures more accurately.

*4) **Inverse identity loss**:* Inverse identity loss is introduced to achieve the opposite effect compared to identity loss, as it attempts to maximize the dissimilarity between two identities. It is defined as follows:

$$\mathcal{L}_{\text{InvID}}\left(x,y\right)=\max\left(0,S_{c}\left(R\big(M(x)\big),R\big(M(y)\big)\right)\right), \tag{11}$$

*5) **MS-SSIM loss**:* MS-SSIM evaluates the structural similarity between images at multiple scales, incorporating variations in image content at different resolutions [36]. This multi-scale approach enables MS-SSIM to capture structural information associated with facial images more robustly and accurately. It has a positive impact on identity reconstruction results for both ReStyle-ID and StyleDemorpher frameworks. The MS-SSIM loss is defined as follows:

$$\mathcal{L}_{\text{MS-SSIM}}\left(x,y\right)=1-\text{MS-SSIM}\left(x,y\right). \tag{12}$$

*6) **ReStyle-ID Training Objective**:* The combined ReStyle-ID training objective consists of four individual loss terms aimed at maximizing the identity similarity between input images and reconstructions. L2, LPIPS, and identity losses are utilized, following the design choices of the ReStyle [11] framework, with an improvement in identity loss through automatic face detection. An additional MS-SSIM loss term is included to further improve identity similarity scores by considering the structural similarity of facial images. The training objective for ReStyle-ID is defined as follows:

$$\begin{aligned}\mathcal{L}_{\text{ReStyle-ID}}\left(I_{x},I_{\hat{y}_{t+1}}\right)&=\lambda_{\text{L2}}\mathcal{L}_{\text{L2}}\left(I_{x},I_{\hat{y}_{t+1}}\right)\\&+\lambda_{\text{LPIPS}}\mathcal{L}_{\text{LPIPS}}\left(I_{x},I_{\hat{y}_{t+1}}\right)\\&+\lambda_{\text{ID}}\mathcal{L}_{\text{ID}}\left(I_{x},I_{\hat{y}_{t+1}}\right)\\&+\lambda_{\text{MS-SSIM}}\mathcal{L}_{\text{MS-SSIM}}\left(I_{x},I_{\hat{y}_{t+1}}\right),\end{aligned} \tag{13}$$

where, I_x is the target image being encoded, $I_{\hat{y}_{t+1}}$ is the reconstructed image at iteration t, and $\lambda_{\text{L2}}=1.0$, $\lambda_{\text{LPIPS}}=0.8$, $\lambda_{\text{ID}}=0.1$, $\lambda_{\text{MS-SSIM}}=0.4$ are the weights scaling the contributions of individual loss functions. These weights have been selected empirically based on identity similarity scores obtained from validation data.

*7) **StyleDemorpher Training Objective**:* The training objective of StyleDemorpher is similar to ReStyle-ID with the addition of a new term corresponding to the inverse identity loss. While the L2, LPIPS, identity, and MS-SSIM loss functions aim to maximize the similarity between the target identity B and the predicted reconstruction $I_{\hat{B}}$, the inverse identity loss is computed between $I_{\hat{B}}$ and $I_{A'}$ to maximize the identity dissimilarity between these images, thereby removing the presence of identity A from the prediction. The training objective for StyleDemorpher is defined as follows:

$$\begin{aligned}\mathcal{L}_{\text{StyleDemorpher}}\left(I_{\hat{B}},\tilde{I}_{B},I_{A'}\right)&=\lambda_{\text{L2}}\mathcal{L}_{\text{L2}}\left(\tilde{I}_{B},I_{\hat{B}}\right)\\&+\lambda_{\text{LPIPS}}\mathcal{L}_{\text{LPIPS}}\left(\tilde{I}_{B},I_{\hat{B}}\right)\\&+\lambda_{\text{ID}}\mathcal{L}_{\text{ID}}\left(\tilde{I}_{B},I_{\hat{B}}\right)\\&+\lambda_{\text{MS-SSIM}}\mathcal{L}_{\text{MS-SSIM}}\left(\tilde{I}_{B},I_{\hat{B}}\right)\\&+\lambda_{\text{InvID}}\mathcal{L}_{\text{InvID}}\left(I_{A'},I_{\hat{B}}\right),\end{aligned} \tag{14}$$

where, $\tilde{I}_B=G(E(I_B))$ corresponds to the image of the target identity B, $I_{A'}$ is the live capture image of the criminal, $I_{\hat{B}}$ is the predicted reconstruction of identity B, and $\lambda_{\text{L2}}=1.0$, $\lambda_{\text{LPIPS}}=0.8$, $\lambda_{\text{ID}}=1.0$, $\lambda_{\text{MS-SSIM}}=0.4$, $\lambda_{\text{InvID}}=0.25$ are the weights scaling the contributions of individual loss functions. These weights have been selected empirically based on identity similarity and dissimilarity scores obtained from validation data. Compared to ReStyle-ID, the weight of the identity loss, λ_{ID}, is increased from 0.1 to 1.0 to better emphasize identity similarity when reconstructing the identity B.

IV. DemorphDB Dataset

This paper introduces the novel DemorphDB dataset, created for training deep learning models to perform face demorphing. DemorphDB is constructed from five datasets comprised of full frontal facial images: FRGC [48], Eurecom-IST Face Dataset [49], Utrecht ECVP Dataset [50], Chicago Face Database [51], [52], [53], and Face Research Lab London Dataset [54]. Images from these datasets have been manually analyzed, retaining only high-quality, passport-like images, excluding those with non-neutral expressions, closed eyes, blurriness, or poor illumination. This resulted in DemorphDB containing images of 1653 unique identities, 643 of which have two or more images.

Apart from bona fide identity images, DemorphDB contains morphs generated automatically for training the StyleDemorpher framework. The images for the morphs are generated following the procedure in Algorithm 1, which uses the notations introduced in Section III-B. This results in a dataset structured into quadruplets of images: (I_B, I_A, $I_{A'}$, I_{AB}).

Algorithm 1 DemorphDB Dataset Construction Procedure

```
Require: DemorphDB bona fide identity images, dlib [55]
    face feature extractor
for each identity $B$ in DemorpDB do
    Find 10 closest identities $A_1, A_2, \ldots, A_{10}$ with at least
        2 available images using dlib face feature extractor
    for each closest identity $A_i$ do
        Randomly select one image as $I_{A_i}$
        Randomly select another image as $I_{A'_i}$ ($I_{A_i} \neq I_{A'_i}$)
    end for
    Select up to 5 images of $B$ ($I_{B_1}, I_{B_2}, \ldots, I_{B_k}$, where
        $k = \min(5, \# \text{ of existing images of } B)$)
    for each image $I_{B_j}$ of $B$ do
        for each closest identity $A_i$ do
            Create morph image $I_{A_i B_j}$ using $I_{A_i}$ and $I_{B_j}$
        end for
    end for
end for
Construct dataset quadruplets $(I_{B_j}, I_{A_i}, I_{A'_i}, I_{A_i B_j})$ for
    each $B_j$ and $A_i$
```

Three types of morphs are available within DemorphDB: UTW [14], UTW-NS, and StyleGAN2. UTW morphs, as described in [14], introduce an automatic method for generating high-quality morphs using splicing [16] technique. This method effectively crops the facial region of the morphed image and pastes it into the image of one of the original identities, removing ghosting artifacts outside the face region. In this work, the cropped face region is pasted into the image of accomplice B, as the accomplice aims to obtain the passport with the morphed image. Additionally, this method warps the geometry of facial parts and then swaps them, such as including the eyes and nose of the criminal while having the mouth of the accomplice.

Due to the swapping of facial parts, UTW morphs remove information about the eyes and nose of the accomplice, which needs to be reconstructed by face demorphing, leaving only their geometry. Therefore, UTW-NS (UTW - No Swapping) morphs are introduced, performed without swapping facial regions, effectively preserving the identity information of both individuals within the morphs.

Finally, StyleGAN2 morphs are also introduced and generated using the ReStyle-ID framework to obtain latent codes of the two identities and then morphing them by averaging:

$$I_{AB} = G\left(\frac{E\left(I_A\right) + E\left(I_B\right)}{2}\right). \tag{15}$$

Overall, DemorphDB contains 36,983 morph images for each of the three discussed morphing methods. Appendix B provides an evaluation of the quality of the morphs, with some examples shown in Figure 7. Notably, all images within this database (bona fide and morphs) have been automatically white balance corrected by a pre-trained network described in [56]. Additionally, all images have been aligned and cropped using the FFHQ method [9].

V. Experiments

A. Datasets

*1) **ReStyle-ID Framework Datasets**:* The original ReStyle [11] framework was trained on the FFHQ [9] dataset, which contains 70,000 images. For training ReStyle-ID, the following datasets are used:

- FFHQ [9] - 70,000 images
- CelebA-HQ [40] train set - 24,000 images
- Synthetic passport-like dataset - 6,652 images

The CelebA-HQ dataset is added to increase the number of unique identities, while the synthetic dataset, generated using StyleGAN2, provides passport-like images for training ReStyle-ID to handle similar images during the demorphing process. To ensure the synthetic StyleGAN2 images have frontal poses and neutral expressions, a pre-trained pSp [30] network is used. This network receives input segmentation masks and creates encodings of random identities in StyleGAN2's latent space. First, segmentation masks are automatically generated using the pre-trained face parsing network from [37] on passport-like images from the DemorphDB dataset. Next, the pSp network generates encodings of random identities with matching segmentation masks. Finally, StyleGAN2 processes these encodings to generate the corresponding images. Figure 8 illustrates some examples of these synthetic images.

Evaluation of the ReStyle-ID framework is performed using images from the DemorphDB dataset. One random image from each of the 1,653 bona fide identities is selected, forming the ***DemorphDB-Single*** evaluation dataset.

*2) **StyleDemorpher Framework Datasets**:* Training the StyleDemorpher framework utilizes quadruplets of images from the DemorphDB dataset. Only UTW-NS and StyleGAN2 morphs are used in training because UTW [14] morphs result in information loss due to swapping of the face parts. However, UTW morphs are included in the evaluation as an unseen morphing method during training.

The subset of target images B from the Face Research Lab London (FRLL) Dataset [54] is reserved for evaluation. Consequently, all morphs with the target demorphing identity B from the FRLL dataset are excluded from training. This results in an evaluation dataset containing 102 unseen target identities. Since the FRLL dataset includes only one neutral expression image per individual, the images for the corresponding identity A (I_A, $I_{A'}$) are still selected based on Algorithm 1.

Additionally, the FRLL-Morphs [57] dataset, based on identities from the FRLL [54] dataset, is used for evaluation to test StyleDemorpher on unseen morphing methods. This dataset includes morphs generated using five different morphing methods: OpenCV [58], FaceMorpher [59], WebMorph [60], AMSL [15], and StyleGAN2 [9], [61]. Since the FRLL dataset lacks multiple images for each identity, $I_{A'}$ images are unavailable. Instead, it is assumed that $I_{A'} = I_A$, meaning the same image used to generate the morph is also used as the live capture image fed to StyleDemorpher.

B. Experimental Setup

The ReStyle-ID framework is trained and evaluated by setting the number of iterations to 5 ($N = 5$), consistent with the methodology described by [11]. Detailed training procedures are provided in Appendix C. The evaluation primarily focuses on the identity similarity between the input images and their encoded reconstructions. Additionally, computational times and the quality of embedding locations within the StyleGAN2 latent space [9] are assessed. Comparative analysis is performed against other state-of-the-art (SOTA) StyleGAN2 embedding methods, including pSp [30], ReStyle [11], and optimization-based approach [9].

For the StyleDemorpher framework, two different versions of the StyleDemorpher networks are trained: one with UTW-NS morphs and the other with StyleGAN2 morphs as described in Section IV. This approach is chosen because traditional and StyleGAN2 morphs differ significantly, with the former using splicing techniques [16] and the latter creating full morphs, including the outside face regions. Training details for this framework are provided in Appendix C.

The primary evaluation of StyleDemorpher focuses on its ability to reconstruct the image of identity B, as this is the main objective of face demorphing. Additional experiments are introduced to assess StyleDemorpher for Differential Morph Attack Detection (DMAD) as illustrated in Figure 10. The generalizability of the approach is also evaluated on previously unseen morph types and various image corruptions that might occur in deployment scenarios. Comparisons are made with the use of no face demorphing and with the Face Demorphing method introduced in [4]. To avoid confusion, the capitalized "Face Demorphing" refers to the approach outlined in [4], while "face demorphing" refers to the general procedure of demorphing where the identity of the accomplice within the morph is reconstructed. For all experiments with Face Demorphing method, a demorphing factor of 0.3 is utilized, following the recommendations in [4]. Additionally, the same dlib [55] automatic face landmark detection model is used for creating UTW [14] and UTW-NS morphs as well as for the Face Demorphing method. This gives Face Demorphing an additional advantage since, in a realistic scenario, it is likely that a different landmark detection mechanism would be used when creating morphs.

Finally, three state-of-the-art face recognition systems (FRS) are used to evaluate the identity similarity scores of both frameworks. These systems are MobileFaceNet [62], ArcFace [39], and CurricularFace [63], offering a range of accuracy levels. The decision thresholds for the similarity scores extracted by the FRS models are set according to Frontex guidelines [64]. Specifically, the thresholds are set at values where the False Acceptance Rate (FAR) is 0.1%, based on the identities of the DemorphDB dataset. These thresholds are specified in Table I.

FRS	Decision Threshold
MobileFaceNet [62]	0.6396
ArcFace [39]	0.4894
CurricularFace [63]	0.2929

Table I: Cosine similarity decision thresholds of FRS models for FAR@0.1% based on identities of the DemorphDB dataset.

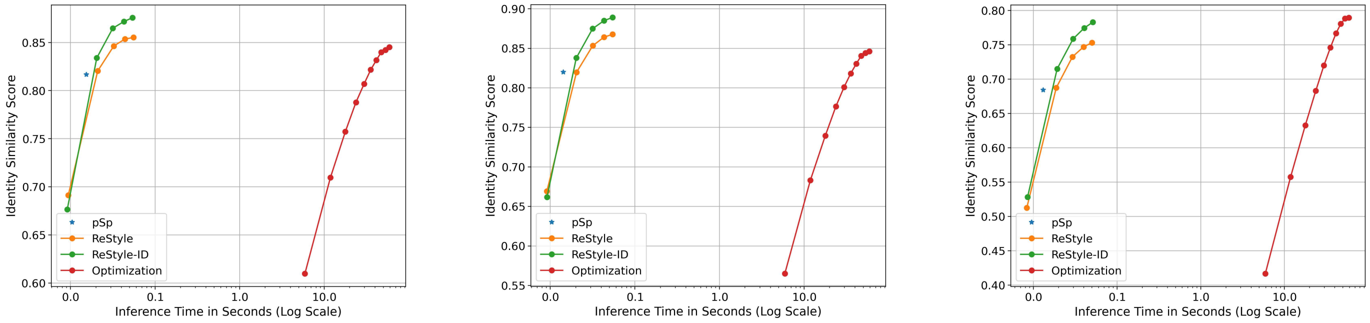
C. Evaluation: ReStyle-ID Framework

*1) **Identity Reconstruction**:* To evaluate the identity reconstruction quality of the ReStyle-ID framework, Table II displays the cosine similarity scores between input identities from the DemorphDB-Single dataset and their reconstructions based on three different FRS models. Across all FRS models, ReStyle-ID similarity scores are higher compared to pSp [30], the original ReStyle framework [11], and optimization-based encoding [9], except for CurricularFace, where optimization-based encoding scores the highest. This indicates that the improvements in ReStyle-ID lead to better identity information preservation, essential for the demorphing task. Figure 4 visualizes these results by plotting iteration-based similarity scores against the average inference time for a single image. While the ReStyle-ID framework takes slightly more time to encode a facial image compared to pSp, it significantly improves the results. In contrast, optimization-based encoding takes about three orders of magnitude longer while resulting in similar or worse identity similarity scores.

*2) **Quality of Encodings**:* To evaluate the quality of the encodings within the latent space, often referred to as "editability," a separate experiment is conducted. Unlike the typical editability of latent codes for changing attributes such as facial expressions [33], this work requires a different kind of editability. The encodings used by StyleDemorpher need to find a different identity within the latent space rather than

Encoding method FSR	pSp [30]	ReStyle [11]	ReStyle-ID	Optimization [9]
MobileFaceNet [62]	0.817 ± 0.073	0.855 ± 0.073	**0.876 ± 0.071**	0.845 ± 0.071
ArcFace [39]	0.820 ± 0.048	0.868 ± 0.049	**0.889 ± 0.043**	0.846 ± 0.048
CurricularFace [63]	0.684 ± 0.058	0.753 ± 0.051	0.783 ± 0.047	**0.789 ± 0.049**

Table II: Identity similarity scores between input identities and their StyleGAN2-encoded reconstructions. The results are presented as mean ± standard deviation. Values in bold signify the best results.



(a) MobileFaceNet [62]

(b) ArcFace [39]

(c) CurricularFace [63]

Figure 4: Identity similarity scores plotted against average inference times for StyleGAN2 identity encoding.

editing the same identity. Therefore, the embeddings must be located in well-defined regions of the StyleGAN2 latent space.

To measure this, the DemorphDB-Single dataset is used to create pairs of the most similar identities. Both images are then encoded into the latent space of StyleGAN2, and morphs are created by averaging (see Equation (15)). The quality of these morphs is evaluated by computing Mated Morph Presentation Match Rate (MMPMR) [65] values between the created morphs and the two original identities. Higher MMPMR values indicate better placement of encodings within the latent space, as the interpolated morphs effectively contain information from both individuals.

Table III displays these MMPMR values. The ReStyle-ID framework scores the highest due to its higher quality of encodings. Interestingly, optimization-based encodings perform the worst, signifying that optimization-based embeddings do not place the embeddings in well-defined regions within the latent space compared to encoder-based methods. Therefore, for the task of face demorphing, encoder architectures provide a faster and more accurate method of finding a well-defined embedding within the StyleGAN2 latent space.

3) ***Morphing Artifact Removal:*** When creating landmark-based morphs, the ghosting artifacts can often occur in resulting morphs due to inaccuracies of landmark detection methods, as well as the availability of only a limited number of landmarks. It has been observed that when encoding morph images with ghosting artifacts, these artifacts are largely suppressed, as shown in Figure 9. This is advantageous for the demorphing procedure because StyleDemorpher will not encounter these artifacts, ensuring they do not have major impact on training.

D. Evaluation: StyleDemorpher Framework

1) ***Visual Comparison:*** The visual results of face demorphing are shown in Figure 13, which presents the demorphing results of StyleDemorpher and Face Demorphing [4] based on UTW, UTW-NS, and StyleGAN2 morphs. Due to licensing restrictions on other datasets used in DemorphDB, only images from the FRLL [54] dataset, which permits publication, are displayed. As can be seen, StyleDemorpher generates accurate reconstructions that closely resemble the accomplice B, while having minimal traces of identity A. In contrast, the Face Demorphing [4] method introduces image artifacts within the facial region. StyleDemorpher, however, generates artifact-free inner face regions, with some artifacts present in the hair, which have minimal impact on identity similarity scores.

2) ***Demorphing Accuracy:*** In this paper, the restoration accuracy from [7], referred to as demorphing accuracy, is used to evaluate the performance of StyleDemorpher. Demorphing accuracy is the percentage of successfully demorphed facial images out of the total number of demorphing attempts. Successful demorphing occurs when the demorphed image $I_{\hat{B}}$ matches I_B but does not match $I_{A'}$ using an FRS decision threshold at FAR@0.1%. Since the decision threshold can vary based on the dataset used for the computation of FAR@0.1%, Figures 5, 14, and 15 plot the demorphing accuracy against different threshold values.

Based on these results, it is evident that compared to the baseline case with no demorphing, where images I_{AB} are used instead of $I_{\hat{B}}$, StyleDemorpher significantly improves results by having higher demorphing accuracy at FAR@0.1% and across a wider range of threshold values. When compared to the results of the Face Demorphing method, a clear improvement is observed with StyleGAN2 morphs. For UTW and UTW-NS morphs, the improvements are less pronounced but still present, as the demorphing accuracy at the FAR@0.1% threshold and its surroundings is higher. The higher demorphing accuracy of StyleDemorpher at lower FRS thresholds indicates that it can be effectively utilized with FRS decision

Encoding method **FSR**	**pSp [30]**	**ReStyle [11]**	**ReStyle-ID**	**Optimization [9]**
MobileFaceNet [62]	39.68%	54.20%	**58.56%**	18.39%
ArcFace [39]	91.53%	94.25%	**95.09%**	68.54%
CurricularFace [63]	96.13%	97.76%	**99.21%**	90.74%

Table III: MMPMR [65] values of the StyleGAN2 morphs generated by encoding two identities and averaging their latent codes. Higher values correspond to higher quality morphs, effectively capturing both identities within a morph image. Values in bold signify the best results.

thresholds calibrated solely on bona fide images. In contrast to the Face Demorphing method, which achieves improved demorphing accuracy at higher decision thresholds, StyleDemorpher performs well even at the FAR@0.1% threshold, which is determined on bona fide images across all three FRS models utilized in this study. This eliminates the need for finding a different decision threshold to achieve better demorphing accuracy for a specific morphing method.

3) ***Histograms:*** Demorphing accuracy does not directly show the relationship between the demorphed images and the two identities within the morphs. For this purpose, the histograms displayed in Figures 6, 16, and 17 are utilized. The histograms plot the identity similarity scores of four image pairs.

- Pair $(\hat{A}, A')$ corresponds to the case when face demorphing is applied to a bona fide image pair, i.e., the document image and the live capture contain the same identity A. Therefore, for this scenario, the demorphed image is labelled as $\hat{A}$.
- Pairs $(\hat{B}, A')$ and $(\hat{B}, B)$ represent the case when face demorphing is applied to a morphed document image. The resulting demorphed image $\hat{B}$ should have a low identity similarity score with A' while having a high identity similarity with B.
- Pair (B, A') represents an impostor pair (no face demorphing) and is added for reference.

It should be noted that when no face demorphing is used, $\hat{A}$ and $\hat{B}$ are replaced by A and AB, respectively.

In cases where no demorphing is used, a large percentage of pairs (AB, A') (purple histograms) are located to the right of the decision threshold, indicating that the criminal could successfully use the morphed document. Conversely, both Face Demorphing and StyleDemorpher methods shift this histogram to the left, preventing the criminal from using the morphed document. Notably, StyleDemorpher shifts this histogram significantly more, regardless of the FRS model.

At the same time, the pairs $(\hat{B}, B)$ (blue histograms) are often slightly shifted to the right by both demorphing methods, indicating that the reconstructions increasingly represent identity B. However, in the case of UTW and UTW-NS morphs with CurricularFace (Figures 6a and 6b), a slight leftward shift can be observed with StyleDemorpher.

While StyleDemorpher effectively removes the presence of identity A from reconstructions of morphed images, it tends to shift the histogram of bona fide image pairs $(\hat{A}, A')$ (green histograms) to the left compared to the no-demorphing case. Although this shift is also present with the Face Demorphing method, it is more pronounced with StyleDemorpher. This indicates that StyleDemorpher negatively impacts bona fide document images. This occurs because StyleDemorpher is trained only with morphed images as inputs and learns to remove the presence of the live capture identity.

Therefore, when operating under the assumption that the input image is a morph, StyleDemorpher can accurately reconstruct the accomplice's identity with minimal traces of the criminal's identity, effectively performing face demorphing. However, if it is unknown whether the input image is a morph, StyleDemorpher can cause false rejections of identities using bona fide document images. A potential solution is to use a different method for morph detection. If morph detection is positive, StyleDemorpher can then provide accurate face demorphing results.

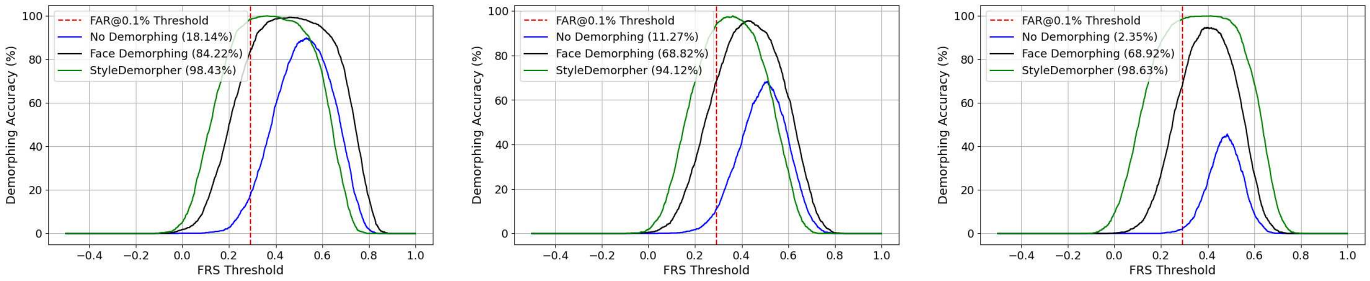
4) ***DMAD Performance:*** To evaluate the Differential Morph Attack Detection (DMAD) performance of StyleDemorpher, the setup described in Appendix F is utilized. Additionally, the following metrics are used:

- Attack Presentation Classification Error Rate (APCER): reports the proportion of incorrectly classified morphing attacks as bona fide representations.
- Bona Fide Presentation Classification Error Rate (BPCER): reports the proportion of incorrectly classified bona fide samples as morphing attacks.
- Detection Equal Error Rate (D-EER): reports the error rate when APCER is equal to BPCER.
- Detection Error Tradeoff (DET) curves: plot APCER values against BPCER values.

Table IV presents the D-EER values along with BPCER values at fixed APCER values of 1%, 5%, and 10%. This table compares the performance of StyleDemorpher with the case of no demorphing and the use of the Face Demorphing [4] method across three FRS models. Figures 18, 19, and 20 plot the DET curves for further visualization.

Based on the results, StyleDemorpher offers the lowest D-EER values when evaluated using MobileFaceNet [62] and ArcFace [39] FRS models. However, the Face Demorphing method performs best with the CurricularFace [63] FRS. This can be explained by the fact that StyleDemorpher, as shown in Section V-D3, has a negative impact when working with bona fide document images. The shift of the bona fide histograms to the left (green histograms in Figures 6, 16, and 17), which is especially pronounced with CurricularFace, negatively impacts the DMAD performance of StyleDemorpher.

Nevertheless, while not specifically designed for DMAD, StyleDemorpher offers an improvement over the baseline when

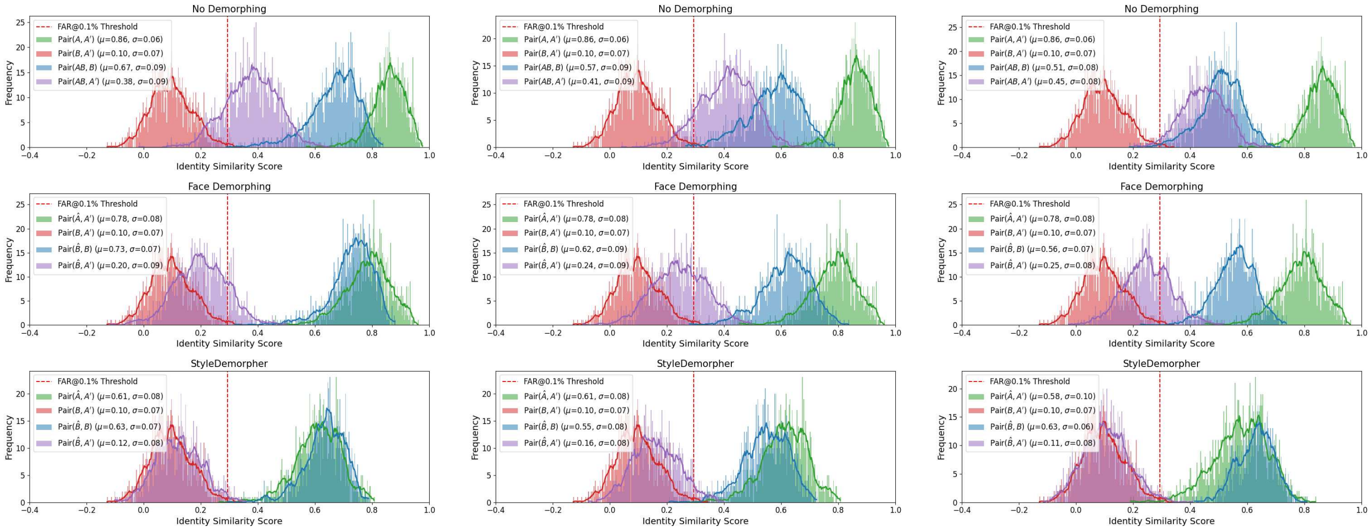


(a) UTW-NS morphs

(b) UTW [14] morphs

(c) StyleGAN2 [9] morphs

Figure 5: Demorphing accuracy plotted against different FRS threshold values of CurricularFace [63]. The dotted red line corresponds to the FAR@0.1% decision threshold. The results are presented for the cases of no demorphing, Face Demorphing [4], and StyleDemorpher face demorphing methods. The values in the brackets correspond to the demorphing accuracy at the FAR@0.1% decision threshold.



(a) UTW-NS morphs

(b) UTW [14] morphs

(c) StyleGAN2 [9] morphs

Figure 6: Histograms visualizing the distributions of the identity similarity scores for different image pairs. The identity scores are computed based on the CurricularFace [63] FRS. The dotted red line corresponds to the FAR@0.1% decision threshold. The results are presented for the cases of no demorphing, Face Demorphing [4], and StyleDemorpher face demorphing methods.

no demorphing is used and provides comparable or better results compared to the Face Demorphing method.

*5) **Generalizability**:* To evaluate how well StyleDemorpher performs on unseen morphing methods, the FRLL-morphs dataset [57], described in Section V-A2, is utilized. Since images $I_{A'}$ corresponding to live capture images are not present in this dataset, images I_A, which were used to generate the morphs, are used instead as one of the inputs for StyleDemorpher (see Figure 3). Due to this reason, the Face Demorphing [4] method has an advantage, as it attempts to invert the morphing process using facial landmarks that were directly employed during the morphing process. Additionally, performance on StyleGAN2 [9], [61] morphs of the FRLL-Morphs dataset is not reported, as these morphs result in low identity similarity scores and are almost always rejected by the three FRS models without any demorphing needed.

Table V reports the demorphing accuracy values at FAR@0.1% for four unseen morphing methods, while Figure 21 visualizes some examples of demorphed images. Based on the results, it can be seen that StyleDemorpher achieves similar or marginally worse demorphing accuracy results compared to the previously seen during training UTW-NS morphs (see Figures 5a, 14a, and 15a). Therefore, it can be concluded that StyleDemorpher generalizes well to previously unseen morphing methods. Moreover, StyleDemorpher outperforms the Face Demorphing [4] method despite the latter's advantage of using images that were directly involved in the morph creation process, and this is especially true for more robust facial recognition systems such as ArcFace [39] and CurricularFace [63].

Finally, Appendix G describes how robust StyleDemorpher is when input images are subjected to different image corruptions. StyleDemorpher displays its ability to generalize well to unseen image corruptions, making it suitable for deployment

Morphing method	**FRS**	**Demorphing method**	**D-EER (%)**	**BPCER @ APCER (%)**		
				1%	**5%**	**10%**
UTW-NS	**MobileFaceNet**	**No Demorphing**	8.73	22.55	11.08	8.43
		Face Demorphing	6.18	19.31	6.86	4.51
		StyleDemorpher	**4.90**	**13.33**	**4.90**	**2.16**
	ArcFace	**No Demorphing**	2.45	3.14	1.77	1.18
		Face Demorphing	1.57	2.16	**0.29**	0.29
		StyleDemorpher	**1.08**	**1.67**	**0.29**	**0.00**
	CurricularFace	**No Demorphing**	0.29	0.20	**0.00**	**0.00**
		Face Demorphing	**0.10**	**0.00**	**0.00**	**0.00**
		StyleDemorpher	0.29	0.20	**0.00**	**0.00**
UTW	**MobileFaceNet**	**No Demorphing**	9.51	23.14	13.04	9.12
		Face Demorphing	6.57	**18.33**	**6.86**	4.71
		StyleDemorpher	**5.88**	19.31	6.96	**3.33**
	ArcFace	**No Demorphing**	2.45	4.51	2.16	1.37
		Face Demorphing	1.86	**2.75**	0.69	0.29
		StyleDemorpher	**1.67**	3.04	**0.39**	**0.20**
	CurricularFace	**No Demorphing**	0.39	0.20	**0.00**	**0.00**
		Face Demorphing	**0.29**	**0.00**	**0.00**	**0.00**
		StyleDemorpher	0.69	0.69	0.10	0.10
StyleGAN2	**MobileFaceNet**	**No Demorphing**	15.78	40.98	27.35	20.59
		Face Demorphing	11.67	34.51	20.20	14.80
		StyleDemorpher	**8.33**	**16.96**	**11.28**	**6.57**
	ArcFace	**No Demorphing**	7.26	17.94	9.12	5.00
		Face Demorphing	4.22	9.41	3.92	2.45
		StyleDemorpher	**2.26**	**6.18**	**0.78**	**0.39**
	CurricularFace	**No Demorphing**	0.39	0.39	**0.00**	**0.00**
		Face Demorphing	**0.30**	**0.00**	**0.00**	**0.00**
		StyleDemorpher	0.69	0.69	0.20	0.20

Table IV: Detection performance of the DMAD methods on UTW-NS, UTW [14], and StyleGAN2 [9] morphs based on the FRLL [54] evaluation subset of the DemorphDB dataset. The results are presented for the cases of no demorphing (using the morphed images directly), Face Demorphing [4], and StyleDemorpher face demorphing methods across three FRS models: MobileFaceNet [62], ArcFace [39], and CurricularFace [63]. Values in bold signify the best results.

Morphing method	**Demorphing method**	**FRS**		
		MobileFaceNet	**ArcFace**	**CurricularFace**
AMSL	**No Demorphing**	59.07%	22.79%	2.26%
	Face Demorphing	**97.28%**	89.83%	68.19%
	StyleDemorpher	95.95%	**99.49%**	**96.78%**
FaceMorpher	**No Demorphing**	23.16%	3.60%	0.49%
	Face Demorphing	85.76%	65.55%	42.80%
	StyleDemorpher	**88.95%**	**95.42%**	**93.13%**
OpenCV	**No Demorphing**	24.16%	3.77%	0.66%
	Face Demorphing	86.16%	61.75%	40.95%
	StyleDemorpher	**88.12%**	**94.68%**	**91.81%**
WebMorph	**No Demorphing**	21.54%	3.52%	0.33%
	Face Demorphing	82.23%	57.66%	37.76%
	StyleDemorpher	**85.42%**	**95.66%**	**90.91%**

Table V: Demorphing accuracy [7] on unseen landmark-based morphing methods from the FRLL-Morphs dataset [57], including AMSL [15], FaceMorpher [59], OpenCV [58], and WebMorph [60]. The results are presented for the cases of no demorphing (using the morphed images directly), Face Demorphing [4], and StyleDemorpher face demorphing methods across three FRS models: MobileFaceNet [62], ArcFace [39], and CurricularFace [63]. Note: images I_A are used instead of $I_{A'}$ due to their unavailability in the FRLL-Morphs dataset. Values in bold signify the best results.

in realistic scenarios where image corruptions could occur.

VI. Conclusion and Future Work

This work introduces a novel deep learning-based face demorphing framework to address limitations observed in current landmark-based and deep learning-based solutions, such as artifacts, low resolution, limited training identities, and poor generalizability [4], [22], [5], [6], [7], [8]. Two interconnected frameworks are presented, collectively aimed at achieving accurate and high-quality face demorphing.

The first framework, ReStyle-ID, builds upon the concepts in [11]. It encodes real facial images into the latent space of StyleGAN2 [9] with minimal loss of identity information. Innovations in the ReStyle-ID framework include removing background distractions to focus the encoder model on identity encoding, employing an automatic face cropping mechanism using a pre-trained MTCNN [35] during identity loss computation, and utilizing the MS-SSIM [36] loss function. The dataset has also been expanded with a mix of real and synthetic images. ReStyle-ID significantly improves identity preservation, placing encodings in well-defined regions of the StyleGAN2 latent space, achieving speeds three orders of magnitude faster than optimization-based approach [9]. This framework acts as the foundational step for face demorphing using StyleGAN2's latent space.

The second framework, StyleDemorpher, is tailored for accurately recovering the identity of accomplices involved in creating morphed document images. A novel dataset, DemorphDB, containing 1653 unique identities, was developed to train this framework. To mitigate overfitting on this relatively small dataset, transfer learning techniques [10] are employed, initializing StyleDemorpher with weights from the pre-trained ReStyle-ID encoder. By processing both the morph image and a trusted live capture image, StyleDemorpher is trained to maximize the resemblance to the target accomplice's identity while minimizing similarity to the criminal identity captured in the live image. It accurately isolates the accomplice's identity from the morph, validating its efficacy in face demorphing. This framework demonstrates high generalizability, performing effectively with novel morphing methods and under various image corruptions.

However, it has been noted that StyleDemorpher adversely affects the analysis of genuine, non-morphed document images. This issue stems from the framework being solely trained on morphed images, presupposing that all processed documents contain morphs. Hence, it is advisable to use StyleDemorpher only after confirming the morphed nature of a document using existing Differential Morph Attack Detection (DMAD) techniques [66], [67], [68]. Future enhancements could include integrating a new subset of genuine paired images into the training process, which would enable StyleDemorpher to minimize its impact on authentic documents, thereby improving overall DMAD performance.

References

[1] *Best Practice Technical Guidelines for Automated Border Control (ABC) Systems*. FRONTEX, 2015. [Online]. Available: https://books.google.nl/books?id=bYONnQAACAAJ

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

[3] R. Raghavendra, K. B. Raja, and C. Busch, "Detecting morphed face images," in *2016 IEEE 8th International Conference on Biometrics Theory, Applications and Systems (BTAS)*, 2016, pp. 1–7.

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

[5] D. Ortega-Delcampo, C. Conde, D. Palacios-Alonso, and E. Cabello, "Border control morphing attack detection with a convolutional neural network de-morphing approach," *IEEE Access*, vol. 8, pp. 92 301–92 313, 2020.

[6] S. Banerjee and A. Ross, "Conditional identity disentanglement for differential face morph detection," in *2021 IEEE International Joint Conference on Biometrics (IJCB)*, 2021, pp. 1–8.

[7] F. Peng, L. bing Zhang, and M. Long, "Fd-gan: Face-demorphing generative adversarial network for restoring accomplice's facial image," 2019.

[8] N. Zhang, X. Liu, X. Li, and G.-J. Qi, "Morphganformer: Transformer-based face morphing and de-morphing," 2023.

[9] T. Karras, S. Laine, M. Aittala, J. Hellsten, J. Lehtinen, and T. Aila, "Analyzing and improving the image quality of stylegan," 2020.

[10] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Transactions on Knowledge and Data Engineering*, vol. 22, no. 10, pp. 1345–1359, 2010.

[11] Y. Alaluf, O. Patashnik, and D. Cohen-Or, "Restyle: A residual-based stylegan encoder via iterative refinement," 2021.

[12] D. F. Rogers and J. A. Adams, *Mathematical Elements for Computer Graphics*, 2nd ed. McGraw-Hill Higher Education, 1989.

[13] M. Ferrara, A. Franco, and D. Maltoni, "Decoupling texture blending and shape warping in face morphing," in *2019 International Conference of the Biometrics Special Interest Group (BIOSIG)*, 2019, pp. 1–5.

[14] I. Batskos and L. Spreeuwers, "Improving fully automated landmark-based face morphing," in *2024 12th International Workshop on Biometrics and Forensics (IWBF)*, 2024, pp. 1–6.

[15] 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, no. 4, pp. 325–332, 2018. [Online]. Available: https://ietresearch.onlinelibrary.wiley.com/doi/abs/10.1049/iet-bmt.2017.0147

[16] A. Makrushin, T. Neubert, and J. Dittmann, "Automatic generation and detection of visually faultless facial morphs," 03 2017.

[17] N. Damer, A. M. Saladié, A. Braun, and A. Kuijper, "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)*, 2018, pp. 1–10.

[18] H. Zhang, S. Venkatesh, R. Ramachandra, K. Raja, N. Damer, and C. Busch, "Mipgan - generating strong and high quality morphing attacks using identity prior driven gan," 2021. [Online]. Available: https://arxiv.org/abs/2009.01729

[19] N. Damer, M. Fang, P. Siebke, J. N. Kolf, M. Huber, and F. Boutros, "Mordiff: Recognition vulnerability and attack detectability of face morphing attacks created by diffusion autoencoders," 2023. [Online]. Available: https://arxiv.org/abs/2302.01843

[20] S. Venkatesh, R. Ramachandra, K. Raja, and C. Busch, "Face morphing attack generation detection: A comprehensive survey," 2020. [Online]. Available: https://arxiv.org/abs/2011.02045

[21] M. Ferrara, A. Franco, and D. Maltoni, *On the Effects of Image Alterations on Face Recognition Accuracy*. Cham: Springer International Publishing, 2016, pp. 195–222. [Online]. Available: https://doi.org/10.1007/978-3-319-28501-6_9

[22] M. Long, J. Zhou, L.-B. Zhang, F. Peng, and D. Zhang, "Adff: Adaptive de-morphing factor framework for restoring accomplice's facial image," *IET Image Processing*, vol. 18, no. 2, pp. 470–480, 2024.

[23] D. A. Hudson and C. L. Zitnick, "Generative adversarial transformers," 2022. [Online]. Available: https://arxiv.org/abs/2103.01209

[24] T. Karras, S. Laine, and T. Aila, "A style-based generator architecture for generative adversarial networks," 2019. [Online]. Available: https://arxiv.org/abs/1812.04948

[25] A. Radford, L. Metz, and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," 2016. [Online]. Available: https://arxiv.org/abs/1511.06434

[26] A. Brock, J. Donahue, and K. Simonyan, "Large scale gan training for high fidelity natural image synthesis," 2019. [Online]. Available: https://arxiv.org/abs/1809.11096

[27] W. Xia, Y. Zhang, Y. Yang, J.-H. Xue, B. Zhou, and M.-H. Yang, "Gan inversion: A survey," 2022. [Online]. Available: https://arxiv.org/abs/2101.05278

[28] R. Abdal, Y. Qin, and P. Wonka, "Image2stylegan: How to embed images into the stylegan latent space?" 2019.

[29] P. Zhu, R. Abdal, Y. Qin, J. Femiani, and P. Wonka, "Improved stylegan embedding: Where are the good latents?" 2021. [Online]. Available: https://arxiv.org/abs/2012.09036

[30] E. Richardson, Y. Alaluf, O. Patashnik, Y. Nitzan, Y. Azar, S. Shapiro, and D. Cohen-Or, "Encoding in style: a stylegan encoder for image-to-image translation," 2021.

[31] O. Tov, Y. Alaluf, Y. Nitzan, O. Patashnik, and D. Cohen-Or, "Designing an encoder for stylegan image manipulation," 2021.

[32] R. Abdal, P. Zhu, N. J. Mitra, and P. Wonka, "Styleflow: Attribute-conditioned exploration of stylegan-generated images using conditional continuous normalizing flows," *ACM Transactions on Graphics*, vol. 40, no. 3, p. 1–21, May 2021. [Online]. Available: http://dx.doi.org/10.1145/3447648

[33] S. Khodadadeh, S. Ghadar, S. Motiian, W.-A. Lin, L. Bölöni, and R. Kalarot, "Latent to latent: A learned mapper for identity preserving editing of multiple face attributes in stylegan-generated images," in *2022 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, 2022, pp. 3677–3685.

[34] M.-H. Le and N. Carlsson, “Styleid: Identity disentanglement for anonymizing faces,” 2022. [Online]. Available: https://arxiv.org/abs/2212.13791

[35] K. Zhang, Z. Zhang, Z. Li, and Y. Qiao, “Joint face detection and alignment using multitask cascaded convolutional networks,” *IEEE Signal Processing Letters*, vol. 23, no. 10, p. 1499–1503, Oct. 2016. [Online]. Available: http://dx.doi.org/10.1109/LSP.2016.2603342

[36] Z. Wang, E. Simoncelli, and A. Bovik, “Multiscale structural similarity for image quality assessment,” in *The Thrity-Seventh Asilomar Conference on Signals, Systems Computers, 2003*, vol. 2, 2003, pp. 1398–1402 Vol.2.

[37] C.-H. Lee, Z. Liu, L. Wu, and P. Luo, “Maskgan: Towards diverse and interactive facial image manipulation,” 2020.

[38] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, “Feature pyramid networks for object detection,” 2017.

[39] J. Deng, J. Guo, J. Yang, N. Xue, I. Kotsia, and S. Zafeiriou, “Arcface: Additive angular margin loss for deep face recognition,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 10, p. 5962–5979, Oct. 2022. [Online]. Available: http://dx.doi.org/10.1109/TPAMI.2021.3087709

[40] T. Karras, T. Aila, S. Laine, and J. Lehtinen, “Progressive growing of gans for improved quality, stability, and variation,” 2018.

[41] A. Dosovitskiy and T. Brox, “Generating images with perceptual similarity metrics based on deep networks,” 2016.

[42] C. Ledig, L. Theis, F. Huszar, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang, and W. Shi, “Photo-realistic single image super-resolution using a generative adversarial network,” 2017.

[43] L. A. Gatys, A. S. Ecker, and M. Bethge, “Image style transfer using convolutional neural networks,” in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 2414–2423.

[44] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, “The unreasonable effectiveness of deep features as a perceptual metric,” 2018.

[45] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “Imagenet classification with deep convolutional neural networks,” in *Advances in Neural Information Processing Systems*, vol. 25. Curran Associates, Inc., 2012.

[46] J. Johnson, A. Alahi, and L. Fei-Fei, “Perceptual losses for real-time style transfer and super-resolution,” 2016.

[47] S. Guan, Y. Tai, B. Ni, F. Zhu, F. Huang, and X. Yang, “Collaborative learning for faster stylegan embedding,” 2020.

[48] P. Phillips, P. Flynn, T. Scruggs, K. Bowyer, J. Chang, K. Hoffman, J. Marques, J. 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, 2005, pp. 947–954 vol. 1.

[49] A. Sepas-Moghaddam, V. Chiesa, P. L. Correia, F. Pereira, and J. Dugelay, “The ist-eurecom light field face database,” in *International Workshop on Biometrics and Forensics, IWBF 2017*, Coventry, UK, April 2017.

[50] P. Hancock, “Psychological image collection at stirling (pics),” 2008. [Online]. Available: http://pics.psych.stir.ac.uk

[51] D. S. Ma, J. Correll, and B. Wittenbrink, “The chicago face database: A free stimulus set of faces and norming data,” *Behavior Research Methods*, vol. 47, pp. 1122–1135, 2015.

[52] D. S. Ma, J. Kantner, and B. Wittenbrink, “Chicago face database: Multiracial expansion,” *Behavior Research Methods*, 2020.

[53] B. Lakshmi, B. Wittenbrink, J. Correll, and D. S. Ma, “The india face set: International and cultural boundaries impact face impressions and perceptions of category membership,” *Frontiers in Psychology*, vol. 12, p. 161, 2020.

[54] L. M. DeBruine and B. C. Jones, “Face research lab london set,” 2017. [Online]. Available: https://api.semanticscholar.org/CorpusID:148812151

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

[56] M. Afifi, B. Price, S. Cohen, and M. S. Brown, “When color constancy goes wrong: Correcting improperly white-balanced images,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 1535–1544.

[57] E. Sarkar, P. Korshunov, L. Colbois, and S. Marcel, “Vulnerability analysis of face morphing attacks from landmarks and generative adversarial networks,” *arXiv preprint*, Oct. 2020. [Online]. Available: https://arxiv.org/abs/2012.05344

[58] S. Mallick. (2016, March) Face morph using opencv — c++ / python — learnopencv. [Online]. [Online]. Available: https://learnopencv.com/face-morph-using-opencv-cpp-python/

[59] A. Quek, “Facemorpher,” 2019.

[60] L. DeBruine, “debruine/webmorph: Beta release 2,” https://doi.org/10.5281/zenodo.1162670, 2018, zenodo.

[61] E. Sarkar, P. Korshunov, L. Colbois, and S. Marcel, “Are gan-based morphs threatening face recognition?” in *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2022, pp. 2959–2963. [Online]. Available: https://doi.org/10.1109/ICASSP43922.2022.9746477

[62] S. Chen, Y. Liu, X. Gao, and Z. Han, “Mobilefacenets: Efficient cnns for accurate real-time face verification on mobile devices,” 2018.

[63] Y. Huang, Y. Wang, Y. Tai, X. Liu, P. Shen, S. Li, J. Li, and F. Huang, “Curricularface: Adaptive curriculum learning loss for deep face recognition,” 2020.

[64] Frontex, *Best practice operational guidelines for Automated Border Control (ABC) systems – Research and development unit*. Publications Office of the European Union, 2012.

[65] 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)*, 2017, pp. 1–7.

[66] U. Scherhag, C. Rathgeb, J. Merkle, and C. Busch, “Deep face representations for differential morphing attack detection,” *IEEE Transactions on Information Forensics and Security*, vol. 15, pp. 3625–3639, 2020.

[67] G. Borghi, E. Pancisi, M. Ferrara, and D. Maltoni, “A double siamese framework for differential morphing attack detection,” *Sensors*, vol. 21, no. 10, 2021. [Online]. Available: https://www.mdpi.com/1424-8220/21/10/3466

[68] B. Chaudhary, P. Aghdaie, S. Soleymani, J. Dawson, and N. M. Nasrabadi, “Differential morph face detection using discriminative wavelet sub-bands,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2021, pp. 1425–1434.

[69] M. R. Zhang, J. Lucas, G. Hinton, and J. Ba, “Lookahead optimizer: k steps forward, 1 step back,” 2019.

[70] L. Liu, H. Jiang, P. He, W. Chen, X. Liu, J. Gao, and J. Han, “On the variance of the adaptive learning rate and beyond,” 2021.

[71] D. Hendrycks and T. Dietterich, “Benchmarking neural network robustness to common corruptions and perturbations,” 2019.

Appendix A
Face Demorphing via StyleGAN2 Latent Codes

The initial design of the StyleDemorpher network aimed to perform face demorphing solely using latent code information, rather than image information. Therefore, in Figure 3, the inputs to the StyleDemorpher network were not the images I_{AB} and $I_{A'}$, but rather their latent encodings $\mathbf{w}_{AB}$ and $\mathbf{w}_{A'}$. Thus, the StyleDemorpher network was a multilayer perceptron (MLP) network rather than a convolutional neural network (CNN). This approach was similar to [33], where the authors used an MLP to edit the latent codes and change identity attributes such as adding facial hair, changing pose, or altering expression.

However, the results of this approach were poor for two reasons. The first and key reason was that the MLP network had to be trained from scratch, resulting in overfitting due to the limited number of unique identities within the DemorphDB dataset. Instead, the current StyleDemorpher framework utilizes the pre-trained ReStyle-ID network's weights as an initial starting point, meaning there is already an established relationship between image and latent spaces based on over 100,000 identities used to train ReStyle-ID.

Secondly, while in [33], the authors performed small edits and largely kept the identity of the person the same, StyleDemorpher requires obtaining a completely different identity. This requires the spatial awareness of CNNs for accurate identity recovery, as the MLP struggles to learn the numerous minute changes within the image space necessary for accurate latent space transformations. Therefore, the MLP-based architecture for facial demorphing based on latent codes proved to be unsuccessful.

Appendix B
DemorphDB Morphs

To evaluate the quality of the morphs in the DemorphDB dataset (see Figure 7 for examples of the morphing methods), the Mated Morph Presentation Match Rate (MMPMR) [65] metric is utilized with the decision threshold of facial recognition systems (FRS) set to a False Acceptance Rate (FAR) of 0.1%. Table VI shows the MMPMR values for the three morphing methods used in the DemorphDB dataset across three different facial recognition systems FRS models. Higher percentage scores indicate higher quality morphs, as they more effectively deceive facial recognition systems into accepting both identities present within a morphed image. As shown, the StyleGAN2 [9] morphs result in the highest scores. This can be explained by StyleGAN2 creating morphs where the region outside the face is also morphed, while UTW [14] and UTW-NS morphs crop the morph and paste it into the image of one of the identities. Thus, UTW and UTW-NS morphs scored higher when accepting the identity whose outer face region matches the morph, while the identity only captured within the inner face part scored lower.

Additionally, different FRS models show varying effectiveness. The simpler and less accurate MobileFaceNet [62] rejects a larger proportion of the morphs, while the more complex and accurate CurricularFace [63] model is often deceived by the morphs. This occurs because, being a better FRS, CurricularFace can detect the traces of both identities used to generate the morph more effectively, making it more susceptible to morphing attacks.

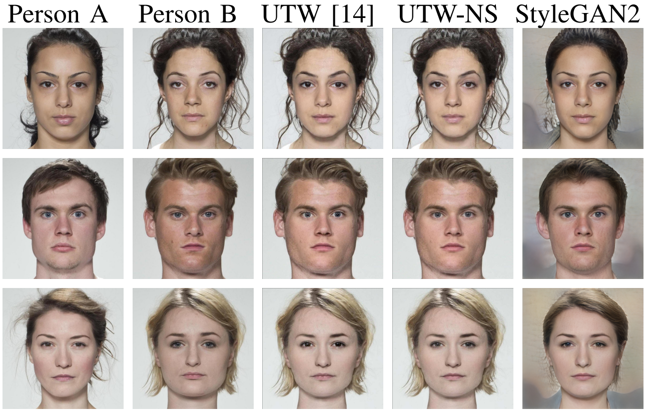


Figure 7: Examples of the morphing methods utilized in the DemorphDB dataset. Person A and B model the identities of the criminal and the accomplice, respectively. UTW [14] and UTW-NS morphs use the splicing technique, while StyleGAN2 [9] morphs also attempt to morph identities outside the face region. Ghosting artifacts are more present in UTW-NS morphs, as they do not swap different face parts between identities like UTW morphs.

Appendix C
Training Details

Both the ReStyle-ID and StyleDemorpher frameworks are trained on input images with 256×256 resolution, while the generated images at the output have 1024×1024 resolution. During the computation of losses, the output images are resized down to 256×256, with the exception of identity (λ_{ID}) and inverse identity (λ_{InvID}) losses, which require an input resolution of 112×112 and further cropping around the face region. The training is performed using the Ranger optimizer, which integrates the Lookahead technique [69] with the Rectified Adam [70] optimizer. A batch size of 6 is utilized, and all experiments are executed on an NVIDIA RTX 4090 GPU.

The ReStyle-ID framework is trained for 18 epochs with a learning rate of 0.0001, while the StyleDemorpher framework (both UTW-NS and StyleGAN2 morph variants) is trained for 20 epochs with a learning rate of 0.00001. Since StyleDemorpher is trained on the DemorphDB dataset with a limited number of target identities, regularization techniques are utilized to prevent overfitting. Weight decay of 0.0001 is applied, and the *map2style* [30] networks (see Figure 2) are modified to include dropout layers. Specifically, 4 dropout layers with a dropout rate of 0.2 are added to each of the 18 *map2style* networks after each Convolution-LeakyReLU block. Finally, it should be noted that only during the training of StyleDemorpher, the input image of identity A is empirically

FRS \ Morphing method	UTW [14]	UTW-NS	StyleGAN2 [9]
MobileFaceNet [62]	21.89%	21.25%	49.59%
ArcFace [39]	63.38%	59.02%	95.41%
CurricularFace [63]	94.37%	90.06%	99.47%

Table VI: MMPMR [65] values for different morphing methods and facial recognition systems. Higher values correspond to higher quality morphs, effectively capturing both identities within a morph image.

set to have a 20% chance to be I_A rather than $I_{A'}$. This is done so that the demorphing network can have an easier understanding of the direct impact of I_A on I_{AB} as well as the indirect relationship between I_A and $I_{A'}$.

Appendix D
Synthetic Passport-like StyleGAN2 Images

Figure 8: Examples of synthetic StyleGAN2 [9] images used for training ReStyle-ID framework.

Appendix E
Morphing Artifact Removal through StyleGAN2 Inversion

Ghosting artifacts often generated when creating landmark-based morphs can pose a problem when performing face demorphing within the latent space of StyleGAN2 [9]. However, since both the StyleGAN2 and ReStyle-ID networks are trained to capture the underlying distribution of typical artifact-free facial images, the encodings are optimized to generate artifact-free images. Therefore, when mapping an image with artifacts into this latent space, the projection is made onto the closest point within the typical learned distribution, resulting in an output without the artifacts.

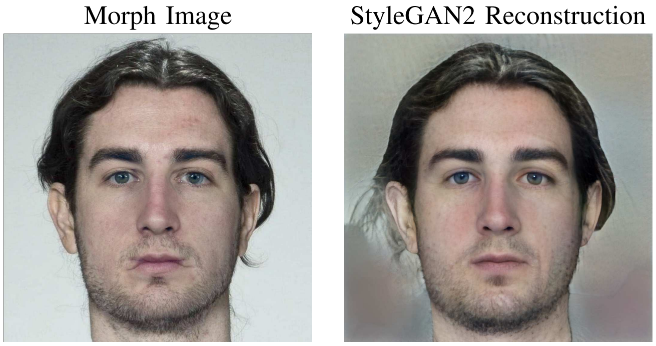


Figure 9: Example of the removal of morphing artifacts when encoding the image into the StyleGAN2 [9] latent space.

Appendix F
Differential Morph Attack Detection with StyleDemorpher

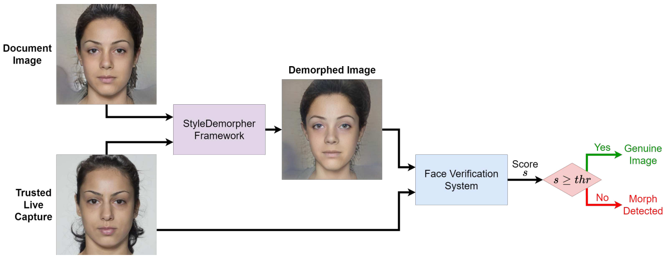


Figure 10: Illustration of the use of StyleDemorpher for Differential Morph Attack Detection.

While the main goal of StyleDemorpher is to accurately reconstruct the identity of the accomplice in the morphed document image, it can also determine if the provided document image is a morph. This capability allows StyleDemorpher to be used for Differential Morph Attack Detection (DMAD). The StyleDemorpher DMAD setup is shown in Figure 10, where given the document image and a trusted live capture, face demorphing is performed using the StyleDemorpher. To decide whether the document image is a morph, the demorphed image is compared to the live capture using a face verification system. If the similarity score is above a certain threshold, the document image is deemed genuine; if the score is low, the document image is considered to be a morph.

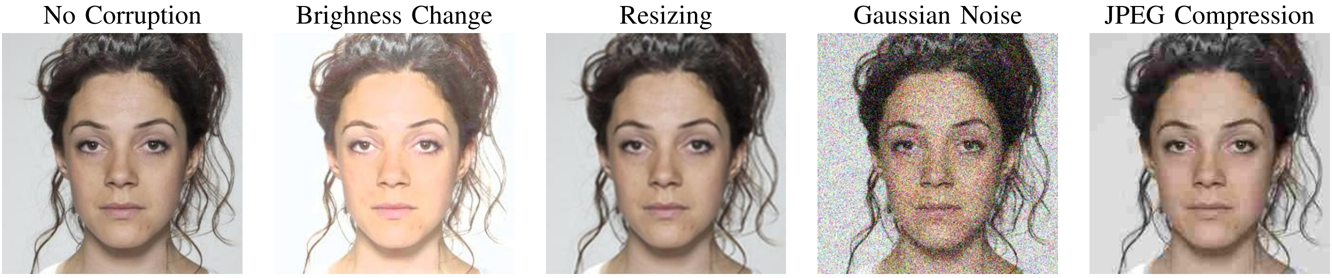


Figure 11: Examples of image corruptions applied to the images before passing them through the StyleDemorpher framework. The brightness change, Gaussian noise, and JPEG compression image corruptions are generated at severity level 3 based on the work of [71]. The resizing image corruption resizes the images from 256×256 down to 128×128, and then back to 256×256, effectively blurring/pixelating the resulting image.

Appendix G
Robustness of StyleDemorpher against Unseen Image Corruptions.

To assess the robustness of the StyleDemorpher against various image distortions, four different types of corruptions are artificially introduced to the input images:

- Brightness change
- Gaussian Noise
- JPEG Compression
- Resizing

The first three corruption types align with the benchmark established by [71], which categorizes multiple artificial image distortions, each with five levels of severity. For this study, a severity level of 3 is selected to mirror more realistic conditions. The remaining resizing corruption involved altering the resolution of the StyleDemorpher's input images from 256×256 to 128×128, and then reverting them back to 256×256. Figure 11 presents visual examples of these image corruptions.

When the corrupted images are generated, the demorphing accuracy curve shown in Figure 12 is plotted to evaluate how each individual image corruption type affects the face demorphing results. It should be noted that to prevent the impacts of FRS models on the evaluation of image corruption robustness, only the input images of the StyleDemorpher Framework were corrupted. This is because the demorphing accuracy metric computes identity similarity scores between the demorphed image $I_{\hat{B}}$ and I_B as well as $I_{A'}$. Therefore, when computing identity similarity scores, non-corrupted versions of I_B and $I_{A'}$ are used, effectively evaluating the quality of the demorphed image $I_{\hat{B}}$ generated from corrupted input images.

Based on the results shown in Figure 12, it can be seen that the majority of image corruptions have minimal effects on the demorphing accuracy, with only brightness change and Gaussian noise image corruptions having any noticeable impact. While the results are only shown for DemorphDB's StyleGAN2 [9] morphs with CurricularFace [63] FRS, similar performance was observed across other morphing methods and FRS models. Therefore, these results show that the StyleDemorpher network is highly resilient to unknown image corruption types.

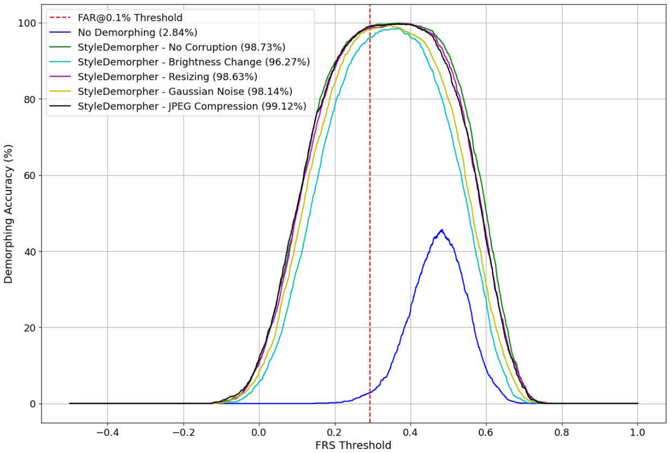


Figure 12: Demorphing accuracy plotted against different FRS threshold values of CirrucularFace [63]. The dotted red line corresponds to the FAR@0.1% decision threshold. The results are presented for the cases of no demorphing, and the use of StyleDemorpher on clean and corrupted by various image corruption methods images. StyleGAN2 [9] morphs of the DemorphDB dataset are utilized. The values in the brackets correspond to the demorphing accuracy at the FAR@0.1% decision threshold.

Appendix H
Additional Figures for StyleDemorpher Framework

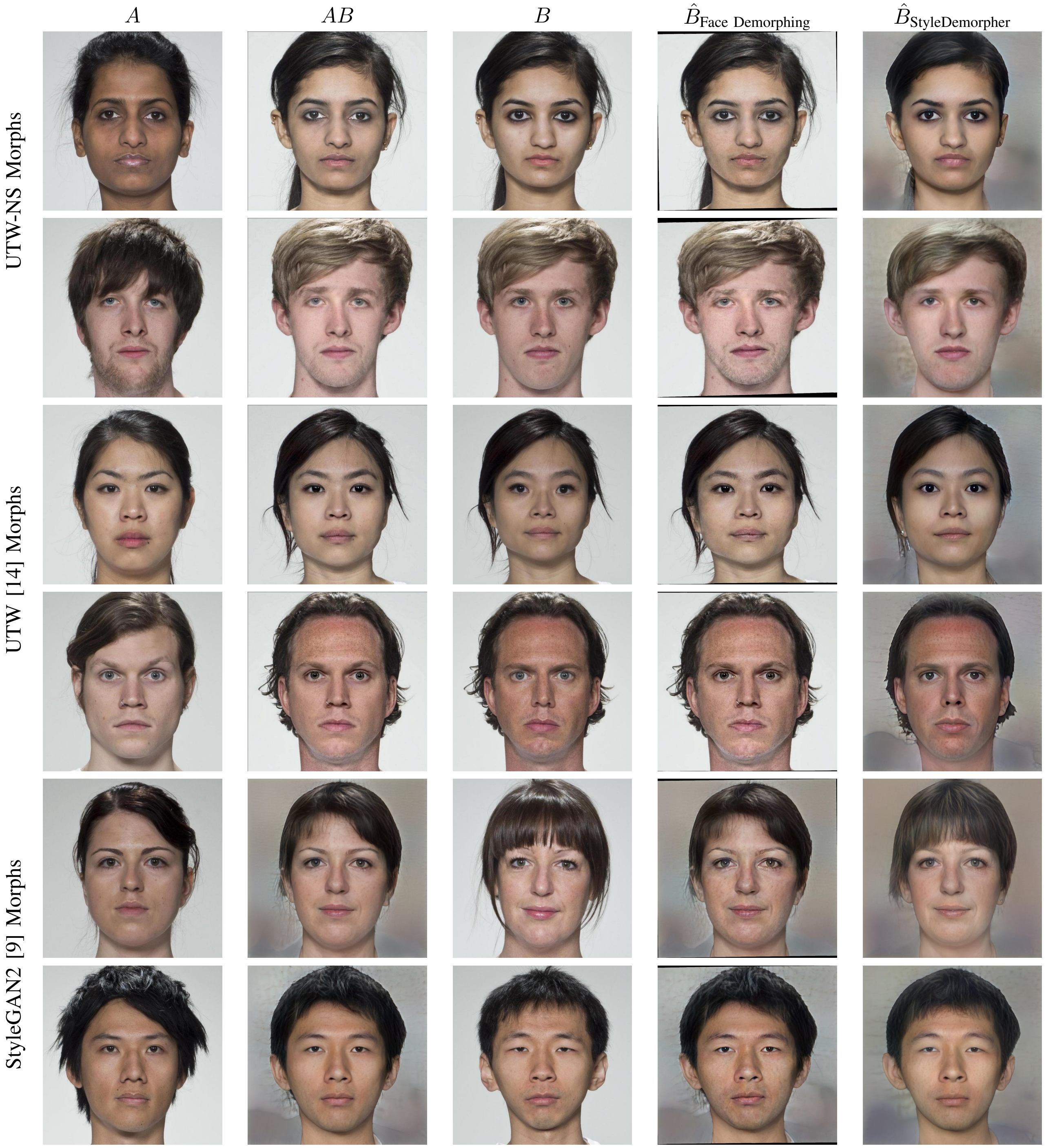
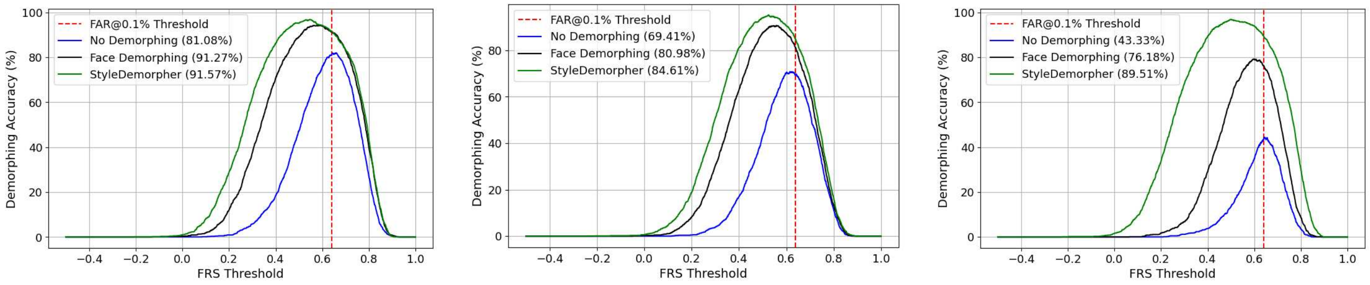

A
AB
B
$\hat{B}_{\text{Face Demorphing}}$
$\hat{B}_{\text{StyleDemorpher}}$
UTW-NS Morphs
UTW [14] Morphs
StyleGAN2 [9] Morphs

Figure 13: Visual results of face demorphing using the FRLL [54] dataset. Due to licensing restrictions, DemorphDB images are not visualized. Since the FRLL dataset lacks trusted live capture images (A'), images directly used to create the morphs (A) are visualized and utilized instead. The goal is to reconstruct image B, representing the accomplice's identity in the morphed image AB. Results from both the StyleDemorpher and Face Demorphing [4] methods are shown.

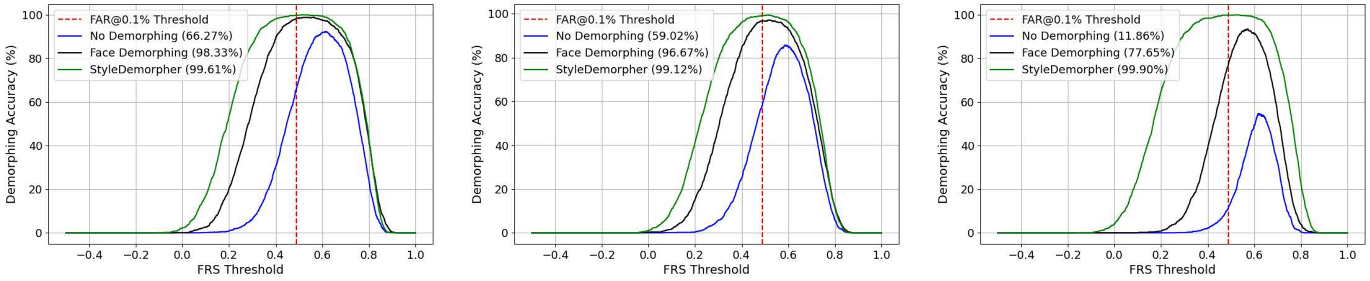


(a) UTW-NS morphs

(b) UTW [14] morphs

(c) StyleGAN2 [9] morphs

Figure 14: Demorphing accuracy plotted against different FRS threshold values of MobileFaceNet [62]. The dotted red line corresponds to the FAR@0.1% decision threshold. The results are presented for the cases of no demorphing, Face Demorphing [4], and StyleDemorpher face demorphing methods. The values in the brackets correspond to the demorphing accuracy at the FAR@0.1% decision threshold.

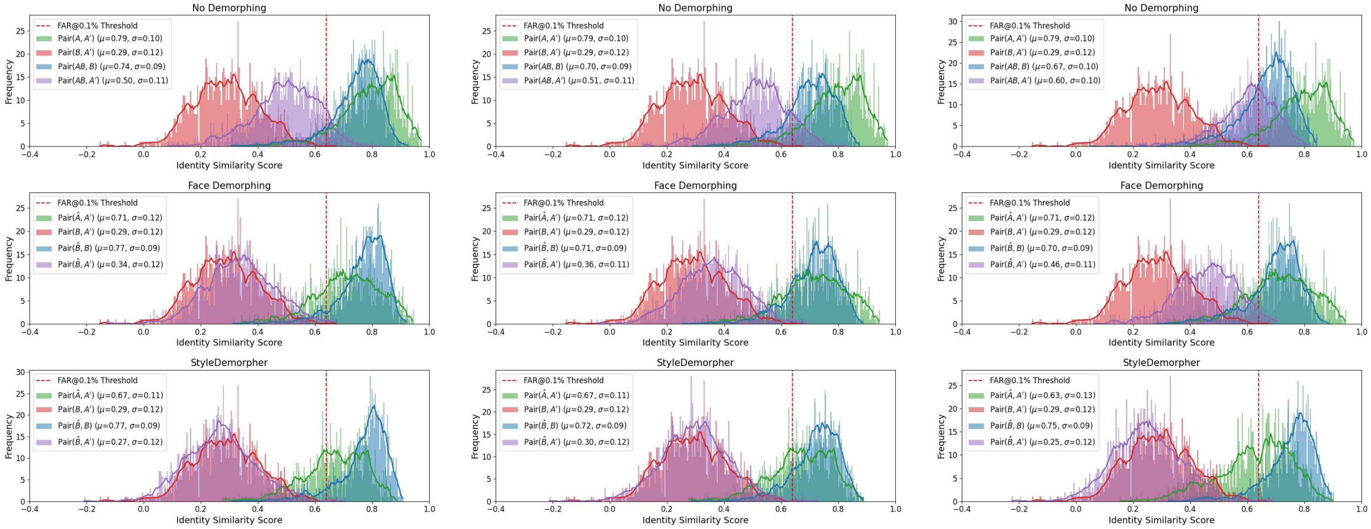


(a) UTW-NS morphs

(b) UTW [14] morphs

(c) StyleGAN2 [9] morphs

Figure 15: Demorphing accuracy plotted against different FRS threshold values of ArcFace [39]. The dotted red line corresponds to the FAR@0.1% decision threshold. The results are presented for the cases of no demorphing, Face Demorphing [4], and StyleDemorpher face demorphing methods. The values in the brackets correspond to the demorphing accuracy at the FAR@0.1% decision threshold.

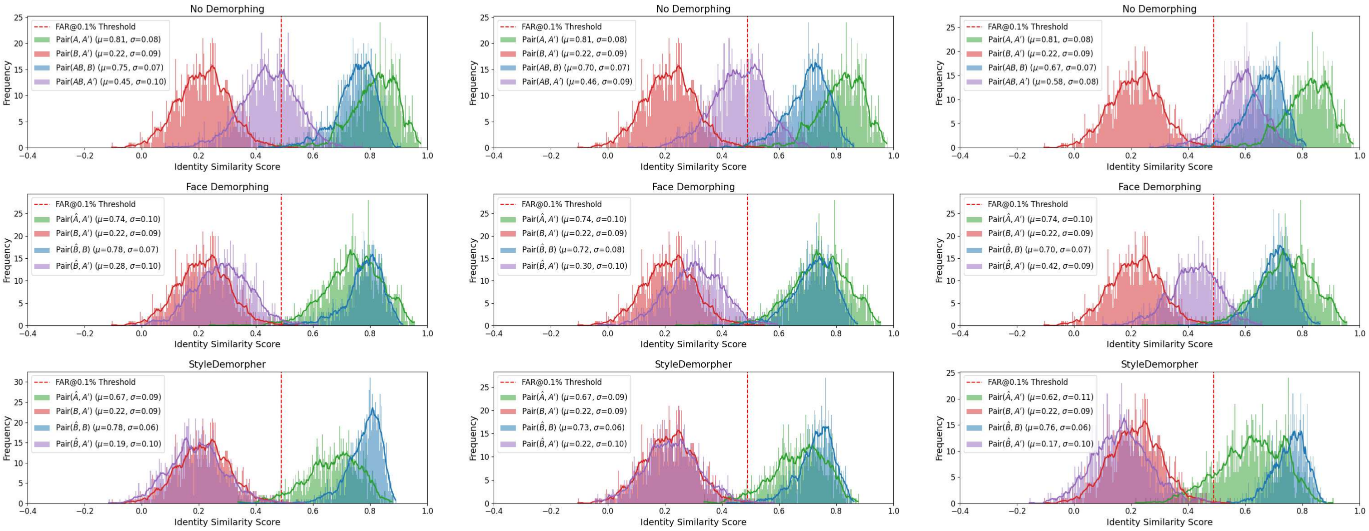


(a) UTW-NS morphs

(b) UTW [14] morphs

(c) StyleGAN2 [9] morphs

Figure 16: Histograms visualizing the distributions of the identity similarity scores for different image pairs. The identity scores are computed based on the MobileFaceNet [62] FRS. The dotted red line corresponds to the FAR@0.1% decision threshold. The results are presented for the cases of no demorphing, Face Demorphing [4], and StyleDemorpher face demorphing methods.

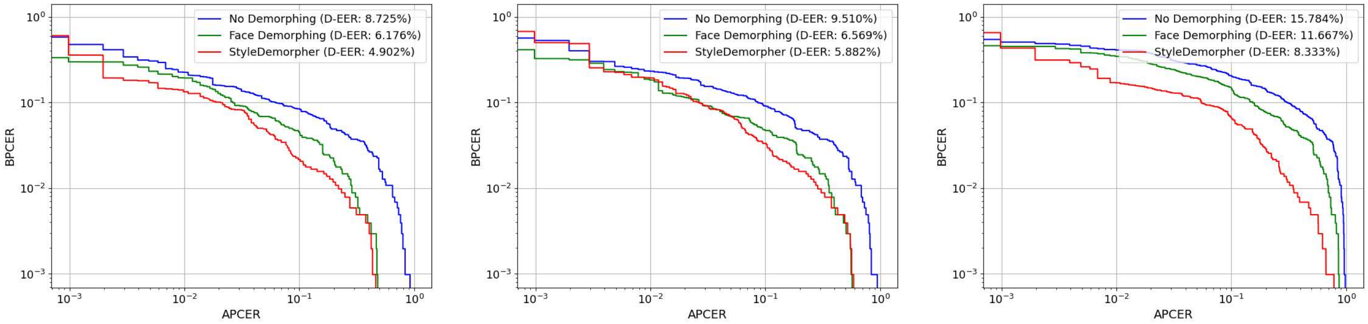


(a) UTW-NS morphs

(b) UTW [14] morphs

(c) StyleGAN2 [9] morphs

Figure 17: Histograms visualizing the distributions of the identity similarity scores for different image pairs. The identity scores are computed based on the ArcFace [39] FRS. The dotted red line corresponds to the FAR@0.1% decision threshold. The results are presented for the cases of no demorphing, Face Demorphing [4], and StyleDemorpher face demorphing methods.

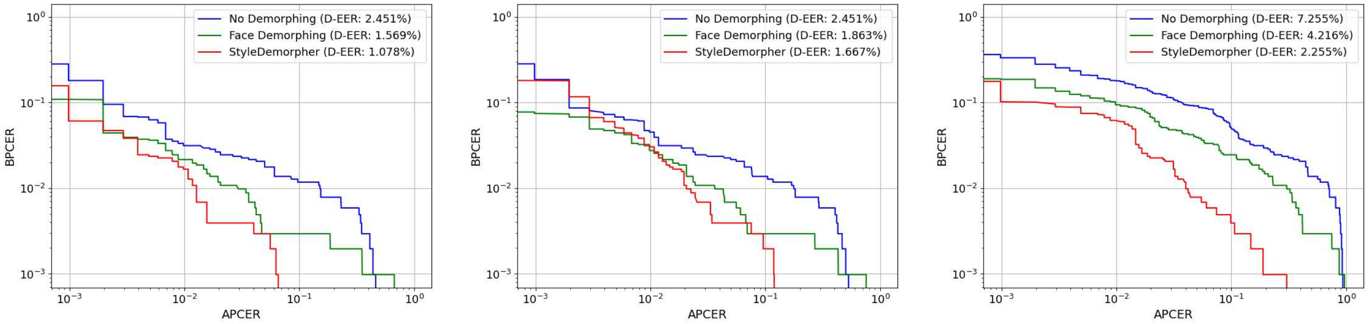


(a) UTW-NS morphs

(b) UTW [14] morphs

(c) StyleGAN2 [9] morphs

Figure 18: Detection Error Tradeoff (DET) curves based on MobileFaceNet [62] FRS. The results are presented for the cases of no demorphing, Face Demorphing [4], and StyleDemorpher face demorphing methods.

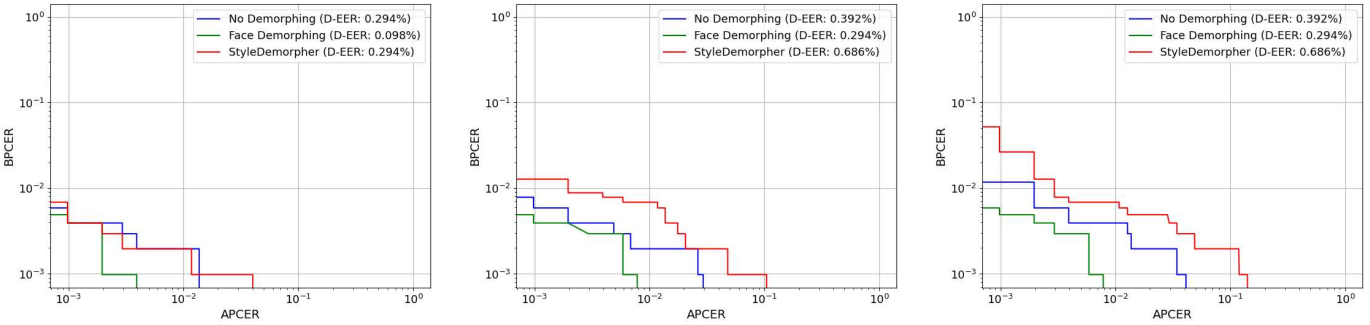


(a) UTW-NS morphs

(b) UTW [14] morphs

(c) StyleGAN2 [9] morphs

Figure 19: Detection Error Tradeoff (DET) curves based on ArcFace [39] FRS. The results are presented for the cases of no demorphing, Face Demorphing [4], and StyleDemorpher face demorphing methods.



(a) UTW-NS morphs

(b) UTW [14] morphs

(c) StyleGAN2 [9] morphs

Figure 20: Detection Error Tradeoff (DET) curves based on CurricularFace [63] FRS. The results are presented for the cases of no demorphing, Face Demorphing [4], and StyleDemorpher face demorphing methods.

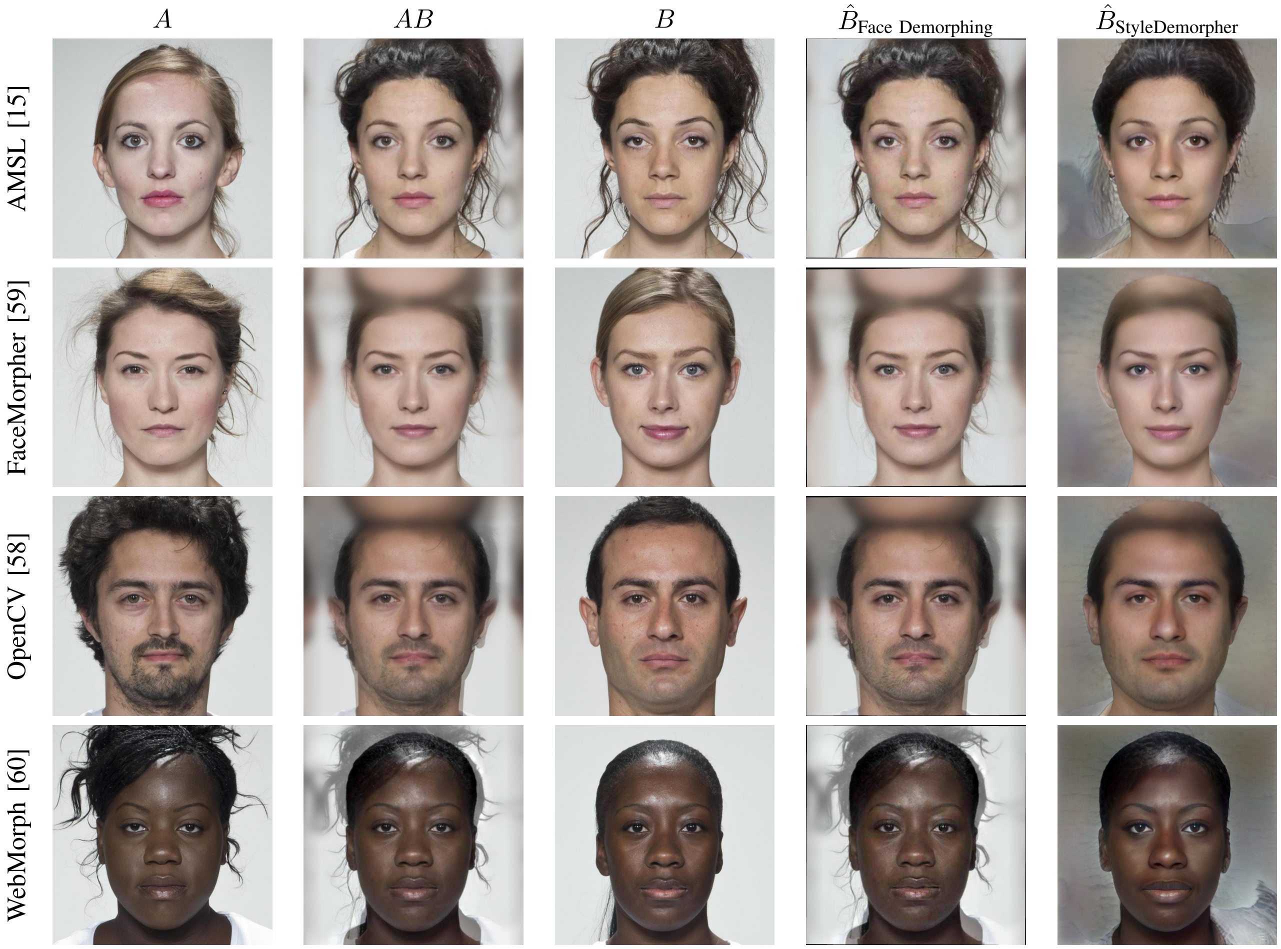

A
AB
B
$\hat{B}_{\text{Face Demorphing}}$
$\hat{B}_{\text{StyleDemorpher}}$
AMSL [15]
FaceMorpher [59]
OpenCV [58]
WebMorph [60]

Figure 21: Visual face demorphing results on the FRLL-Morphs [57] dataset. Due to the lack of $I_{A'}$ images within this dataset, images I_A, which were used in the morph creation, are utilized directly. Results from both StyleDemorpher and the Face Demorphing [4] method are presented. The four presented landmark-based morphing methods were not seen during the training of StyleDemorpher. Note: some of the morph images contain image reflection artifacts introduced during alignment with the FFHQ method [9]. These artifacts, which occur due to the original morphs being excessively zoomed in, are present on the borders of the images as the outer regions of the faces are not captured.