FSGAN: Subject Agnostic Face Swapping and Reenactment

Yuval Nirkin
Bar-Ilan University, Israel
yuval.nirkin@gmail.com

Yosi Keller
Bar-Ilan University, Israel
yosi.keller@gmail.com

Tal Hassner
The Open University of Israel, Israel
talhassner@gmail.com

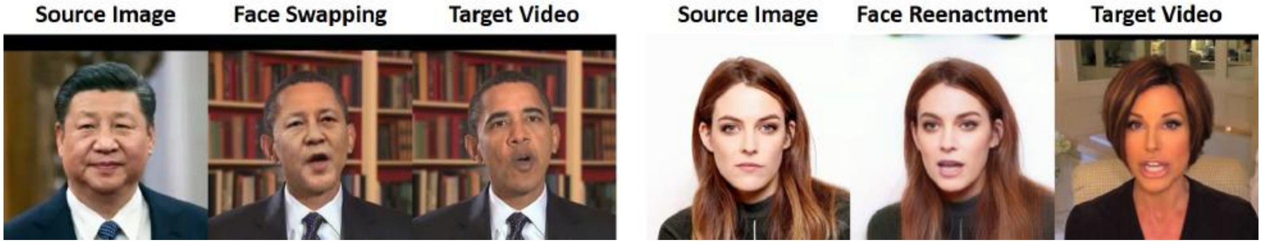

Figure 1: *Face swapping and reenactment.* Left: Source face swapped onto target. Right: Target video used to control the expressions of the face appearing in the source image. In both cases, our results appears in the middle. For more information please visit our website: `https://nirkin.com/fsgan`.

Abstract

We present Face Swapping GAN (FSGAN) for face swapping and reenactment. Unlike previous work, FSGAN is subject agnostic and can be applied to pairs of faces without requiring training on those faces. To this end, we describe a number of technical contributions. We derive a novel recurrent neural network (RNN)–based approach for face reenactment which adjusts for both pose and expression variations and can be applied to a single image or a video sequence. For video sequences, we introduce continuous interpolation of the face views based on reenactment, Delaunay Triangulation, and barycentric coordinates. Occluded face regions are handled by a face completion network. Finally, we use a face blending network for seamless blending of the two faces while preserving target skin color and lighting conditions. This network uses a novel Poisson blending loss which combines Poisson optimization with perceptual loss. We compare our approach to existing state-of-the-art systems and show our results to be both qualitatively and quantitatively superior.

1. Introduction

Face swapping is the task of transferring a face from source to target image, so that it seamlessly replaces a face appearing in the target and produces a realistic result (Fig. 1 left). *Face reenactment* (aka *face transfer* or *puppeteering*) uses the facial movements and expression deformations of a control face in one video to guide the motions and deformations of a face appearing in a video or image (Fig. 1 right). Both tasks are attracting significant research attention due to their applications in entertainment [1, 21, 48], privacy [6, 26, 32], and training data generation.

Previous work proposed either methods for swapping or for reenactment but rarely both. Earlier methods relied on underlying 3D face representations [46] to transfer or control facial appearances. Face shapes were either estimated from the input image [44, 42, 35] or were fixed [35]. The 3D shape was then aligned with the input images [10] and used as a proxy when transferring intensities (swapping) or controlling facial expression and viewpoints (reenactment).

Recently, deep network–based methods were proposed for face manipulation tasks. Generative adversarial networks (GANs) [13], for example, were shown to successfully generate realistic images of fake faces. Conditional GANs (cGANs) [31, 17, 47] were used to transform an image depicting real data from one domain to another and inspired multiple face reenactment schemes [37, 50, 40]. Finally, the DeepFakes project [12] leveraged cGANs for face swapping in videos, making swapping widely accessible to non-experts and receiving significant public attention. Those methods are capable of generating realistic face images by replacing the classic graphics pipeline. They all, however, still implicitly use 3D face representations.

Some methods relied on latent feature space domain separation [45, 34, 33]. These methods decompose the identity component of the face from the remaining traits, and encode identity as the manifestation of latent feature vectors, resulting in significant information loss and limiting the quality of the synthesized images. Subject specific methods [42, 12, 50, 22] must be trained for each subject or pair of subjects and so require expensive subject specific data—typically thousands of face images—to achieve reasonable results, limiting their potential usage. Finally, a major concern shared by previous face synthesis schemes, particularly the 3D based methods, is that they all require special care when handling partially occluded faces.

We propose a deep learning–based approach to face swapping and reenactment in images and videos. Unlike previous work, our approach is *subject agnostic*: it can be applied to faces of different subjects without requiring subject specific training. Our Face Swapping GAN (FSGAN) is end-to-end trainable and produces photo realistic, temporally coherent results. We make the following contributions:

- **Subject agnostic swapping and reenactment.** To the best of our knowledge, our method is the first to simultaneously manipulate pose, expression, and identity without requiring person-specific or pair-specific training, while producing high quality and temporally coherent results.
- **Multiple view interpolation.** We offer a novel scheme for interpolating between multiple views of the same face in a continuous manner based on reenactment, Delaunay Triangulation and barycentric coordinates.
- **New loss functions.** We propose two new losses: A stepwise consistency loss, for training face reenactment progressively in small steps, and a Poisson blending loss, to train the face blending network to seamlessly integrate the source face into its new context.

We test our method extensively, reporting qualitative and quantitative ablation results and comparisons with state of the art. The quality of our results surpasses existing work even without training on subject specific images.

2. Related work

Methods for manipulating the appearances of face images, particularly for face swapping and reenactment, have a long history, going back nearly two decades. These methods were originally proposed due to privacy concerns [6, 26, 32] though they are increasingly used for recreation [21] or entertainment (e.g., [1, 48]).

3D based methods. The earliest swapping methods required manual involvement [6]. An automatic method was proposed a few years later [4]. More recently, Face2Face transferred expressions from source to target face [44]. Transfer is performed by fitting a 3D morphable face model (3DMM) [5, 7, 11] to both faces and then applying the expression components of one face onto the other with care given to interior mouth regions. The reenactement method of Suwajanakorn et al. [42] synthesized the mouth part of the face using a reconstructed 3D model of (former president) Obama, guided by face landmarks, and using a similar strategy for filling the face interior as in Face2Face. The expression of frontal faces was manipulated by Averbuch-Elor et al. [3] by transferring the mouth interior from source to target image using 2D wraps and face landmarks.

Finally, Nirkin et al. [35] proposed a face swapping method, showing that 3D face shape estimation is unnecessary for realistic face swaps. Instead, they used a fixed 3D face shape as the proxy [14, 29]. Like us, they proposed a face segmentation method, though their work was not end-to-end trainable and required special attention to occlusions. We show our results to be superior than theirs.

GAN-based methods. GANs [13] were shown to generate fake images with the same distribution as a target domain. Although successful in generating realistic appearances, training GANs can be unstable and restricts their application to low-resolution images. Subsequent methods, however, improved the stability of the training process [28, 2]. Karras et al. [20] train GANs using a progressive multiscale scheme, from a low to high image resolutions. CycleGAN [52] proposed a cycle consistency loss, allowing training of unsupervised generic transformations between different domains. A cGAN with L_1 loss was applied by Isola et al. [17] to derive the pix2pix method, and was shown to produce appealing synthesis results for applications such as transforming edges to faces.

Facial manipulation using GANs. Pix2pixHD [47] used GANs for high resolution image-to-image translation by applying a multi-scale cGAN architecture and adding a perceptual loss [18]. GANimation [37] proposed a dual generator cGAN conditioned on emotion action units, that generates an attention map. This map was used to interpolate between the reenacted and original images, to preserve the background. GANnotation [40] proposed deep facial reenactment driven by face landmarks. It generates images progressively using a triple consistency loss: it first frontalizes an image using landmarks then processes the frontal face.

Kim et al. [22] recently proposed a hybrid 3D/deep method. They render a reconstructed 3DMM of a specific subject using a classic graphic pipeline. The rendered image is then processed by a generator network, trained to map synthetic views of each subject to photo-realistic images.

Finally, feature disentanglement was proposed as a means for face manipulation. RSGAN [34] disentangles the latent representations of face and hair whereas FSNet [33] proposed a latent space which separates identity and geometric components, such as facial pose and expression.

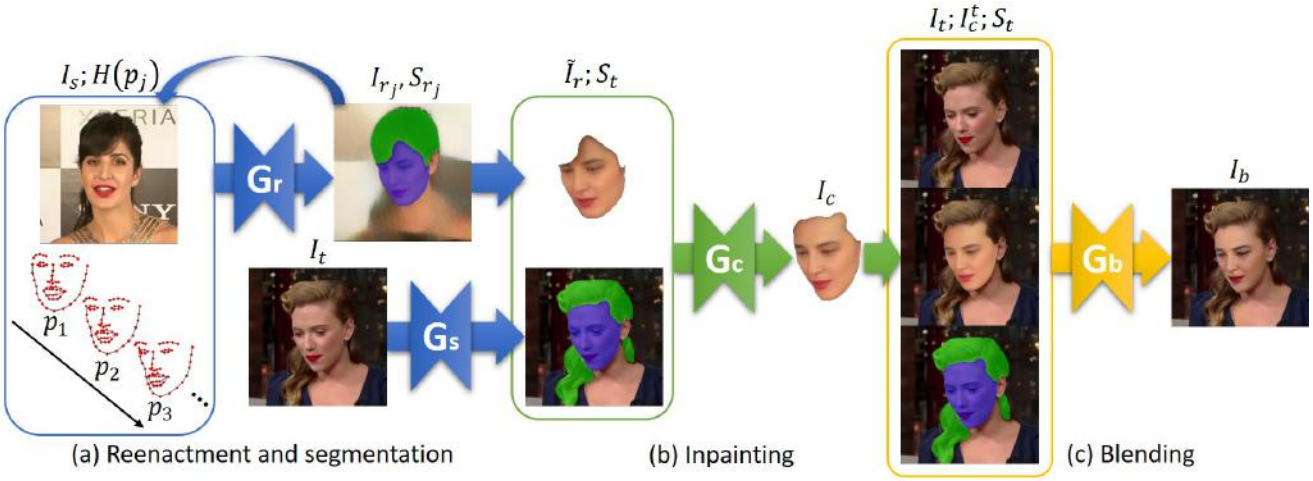

Figure 2: Overview of the proposed FSGAN approach. (a) The recurrent reenactment generator G_r and the segmentation generator G_s. G_r estimates the reenacted face F_r and its segmentation S_r, while G_s estimates the face and hair segmentation mask S_t of the target image I_t. (b) The inpainting generator G_c inpaints the missing parts of $\tilde{F}_r$ based on S_t to estimate the complete reenacted face F_c. (c) The blending generator G_b blends F_c and F_t, using the segmentation mask S_t.

3. Face swapping GAN

In this work we introduce the Face Swapping GAN (FSGAN), illustrated in Fig. 2. Let I_s be the source and I_t the target images of faces $F_s \in I_s$ and $F_t \in I_t$, respectively. We aim to create a new image based on I_t, where F_t is replaced by F_s while retaining the same pose and expression.

FSGAN consists of three main components. The first, detailed in Sec. 3.2 (Fig. 2(a)), consists of a reenactment generator G_r and a segmentation CNN G_s. G_r is given a heatmaps encoding the facial landmarks of F_t, and generates the reenacted image I_r, such that F_r depicts F_s at the same pose and expression of F_t. It also computes S_r: the segmentation mask of F_r. Component G_s computes the face and hair segmentations of F_t.

The reenacted image, I_r, may contain missing face parts, as illustrated in Fig. 2 and Fig. 2(b). We therefore apply the face inpainting network, G_c, detailed in Sec. 3.4 using the segmentation S_t, to estimate the missing pixels. The final part of the FSGAN, shown in Fig. 2(c) and Sec. 3.5, is the blending of the completed face F_c into the target image I_t to derive the final face swapping result.

The architecture of our face segmentation network, G_s, is based on U-Net [38], with bilinear interpolation for upsampling. All our other generators—G_r, G_c, and G_b—are based on those used by pix2pixHD [47], with coarse-to-fine generators and multi-scale discriminators. Unlike pix2pixHD, our global generator uses a U-Net architecture with bottleneck blocks [15] instead of simple convolutions and summation instead of concatenation. As with the segmentation network, we use bilinear interpolation for upsampling in both global generator and enhancers. The actual number of layers differs between generators.

Following others [50], training subject agnostic face reenactment is non-trivial and might fail when applied to unseen face images related by large poses. To address this challenge, we propose to break large pose changes into small manageable steps and interpolate between the closest available source images corresponding to a target's pose. These steps are explained in the following sections.

3.1. Training losses

Domain specific perceptual loss. To capture fine facial details we adopt the perceptual loss [18], widely used in recent work for face synthesis [40], outdoor scenes [47], and super resolution [25]. Perceptual loss uses the feature maps of a pretrained VGG network, comparing high frequency details using a Euclidean distance.

We found it hard to fully capture details inherent to face images, using a network pretrained on a generic dataset such as ImageNet. Instead, our network is trained on the target domain: We therefore train multiple VGG-19 networks [41] for face recognition and face attribute classification. Let $F_i \in \mathbb{R}^{C_i \times H_i \times W_i}$ be the feature map of the i-th layer of our network, the perceptual loss is given by

$$\mathcal{L}_{perc}(x, y) = \sum_{i=1}^{n} \frac{1}{C_i H_i W_i} \left\| F_i(x) - F_i(y) \right\|_1 . \qquad (1)$$

Reconstruction loss. While the perceptual loss of Eq. (1) captures fine details well, generators trained using only that loss, often produce images with inaccurate colors, corresponding to reconstruction of low frequency image content. We hence also applied a pixelwise L_1 loss to the generators:

$$\mathcal{L}_{pixel}(x, y) = \|x - y\|_1 . \qquad (2)$$

The overall loss is then given by

$$\mathcal{L}_{rec}(x,y) = \lambda_{perc}\mathcal{L}_{perc}(x,y) + \lambda_{pixel}\mathcal{L}_{pixel}(x,y). \quad (3)$$

The loss in Eq. (3) was used with all our generators.

Adversarial loss. To further improve the realism of our generated images we use an adversarial objective [47]. We utilized a multi-scale discriminator consisting of multiple discriminators, $D_1, D_2, ..., D_n$, each one operating on a different image resolution. For a generator G and a multi-scale discriminator D, our adversarial loss is defined by:

$$\mathcal{L}_{adv}(G,D) = \min_{G} \max_{D_1,\dots D_n} \sum_{i=1}^{n} \mathcal{L}_{GAN}(G, D_i), \quad (4)$$

where $\mathcal{L}_{GAN}(G,D)$ is defined as:

$$\begin{aligned}\mathcal{L}_{GAN}(G,D) =& \mathbb{E}_{(x,y)}[\log D(x,y)] \\ &+ \mathbb{E}_{x}[\log(1 - D(x, G(x)))]. \quad (5)\end{aligned}$$

3.2. Face reenactment and segmentation

Given an image $I \in \mathbb{R}^{3\times H\times W}$ and a heatmap representation $H(p) \in \mathbb{R}^{70\times H\times W}$ of facial landmarks, $p \in \mathbb{R}^{70\times 2}$, we define the face reenactment generator, G_r, as the mapping $G_r : \left\{\mathbb{R}^{3\times H\times W}, \mathbb{R}^{70\times H\times W}\right\} \rightarrow \mathbb{R}^{3\times H\times W}$.

Let $v_s, v_t \in \mathbb{R}^{70\times 3}$ and $e_s, e_t \in \mathbb{R}^{3}$, be the 3D landmarks and Euler angles corresponding to F_s and F_t. We generate intermediate 2D landmark positions p_j by interpolating between e_s and e_t, and the centroids of v_s and v_t, using intermediate points for which we project v_s back to I_s. We define the reenactment output recursively for each iteration $1 \leq j \leq n$ as

$$I_{r_j}, S_{r_j} = G_r(I_{r_{j-1}}; H(p_j)), \quad (6)$$

$$I_{r_0} = I_s.$$

Similar to others [37], the last layer of the global generator and each of the enhancers in G_r is split into two heads: the first produces the reenacted image and the second the segmentation mask. In contrast to binary masks used bu others [37], we consider the face and hair regions separately. The binary mask implicitly learned by the reenactment network captures most of the head including the hair, which we segment separately. Moreover, the additional hair segmentation also improves the accuracy of the face segmentation. The face segmentation generator G_s is defined as $G_r : \mathbb{R}^{3\times H\times W} \rightarrow \mathbb{R}^{3\times H\times W}$, where given an RGB image it output a 3-channels segmentation mask encoding the background, face, and hair.

Training. Inspired by the triple consistency loss [40], we propose a stepwise consistency loss. Given an image pair (I_s, I_t) of the same subject from a video sequence, let I_{r_n} be the reenactment result after n iterations, and $\widetilde{I}_t, \widetilde{I}_{r_n}$ be the same images with their background removed using the segmentation masks S_t and S_{r_j}, respectively. The stepwise consistency loss is defined as: $\mathcal{L}_{rec}(\widetilde{I}_{r_n}, \widetilde{I}_t)$. The final objective for the G_r:

$$\begin{aligned}\mathcal{L}(G_r) =& \lambda_{stepwise}\mathcal{L}_{rec}(\widetilde{I}_{r_n}, \widetilde{I}_t) + \lambda_{rec}\mathcal{L}_{rec}(\widetilde{I}_r, \widetilde{I}_t) \\ &+ \lambda_{adv}\mathcal{L}_{adv} + \lambda_{seg}\mathcal{L}_{pixel}(S_r, S_t). \quad (7)\end{aligned}$$

For the objective of G_s we use the standard cross-entropy loss, L_{ce}, with additional guidance from G_r:

$$\mathcal{L}(G_s) = L_{ce} + \lambda_{reenactment}\mathcal{L}_{pixel}(S_t, S_r^t), \quad (8)$$

where S_r^t is the segmentation mask result of $G_r(I_t; H(p_t))$ and p_t is the 2D landmarks corresponding to I_t.

We train both G_r and G_s together, in an interleaved fashion. We start with training G_s for one epoch followed by the training of G_r for an additional epoch, increasing $\lambda_{reenactment}$ as the training progresses. We have found that training G_r and G_s together helps filtering noise learned from coarse face and hair segmentation labels.

3.3. Face view interpolation

Standard computer graphics pipelines project textured mesh polygons onto a plane for seamless rendering [16]. We propose a novel, alternative scheme for continuous interpolation between face views. This step is an essential phase of our method, as it allows using the entire source video sequence, without training our model on a particular video frame, making it subject agnostic.

Given a set of source subject images, $\{\mathbf{I}_{s_1}, \dots, \mathbf{I}_{s_n}\}$, and Euler angles, $\{\mathbf{e}_1, \dots, \mathbf{e}_n\}$, of the corresponding faces $\{\mathbf{F}_{s_1}, \dots, \mathbf{F}_{s_n}\}$, we construct the appearance map of the source subject, illustrated in Fig. 3(a). This appearance map embeds head poses in a triangulated plane, allowing head poses to follow continuous paths.

We start by projecting the Euler angles $\{\mathbf{e}_1, \dots, \mathbf{e}_n\}$ onto a plane by dropping the roll angle. Using a k-d tree data structure [16], we remove points in the angular domain that are too close to each other, prioritizing the points for which the corresponding Euler angles have a roll angle closer to zero. We further remove motion blurred images. Using the remaining points, $\{x_1, \dots, x_m\}$, and the four boundary points, $y_i \in [-75, 75] \times [-75, 75]$, we build a mesh, M, in the angular domain by Delaunay Triangulation.

For a query Euler angle, e_t, of a face, F_t, and its corresponding projected point, x_t, we find the triangle $T \in M$ that contains x_t. Let $x_{i_1}, x_{i_2}, x_{i_3}$ be the vertices of T and $I_{s_{i_1}}, I_{s_{i_2}}, I_{s_{i_3}}$ be the corresponding face views. We calculate the barycentric coordinates, $\lambda_1, \lambda_2, \lambda_3$ of x_t, with re-

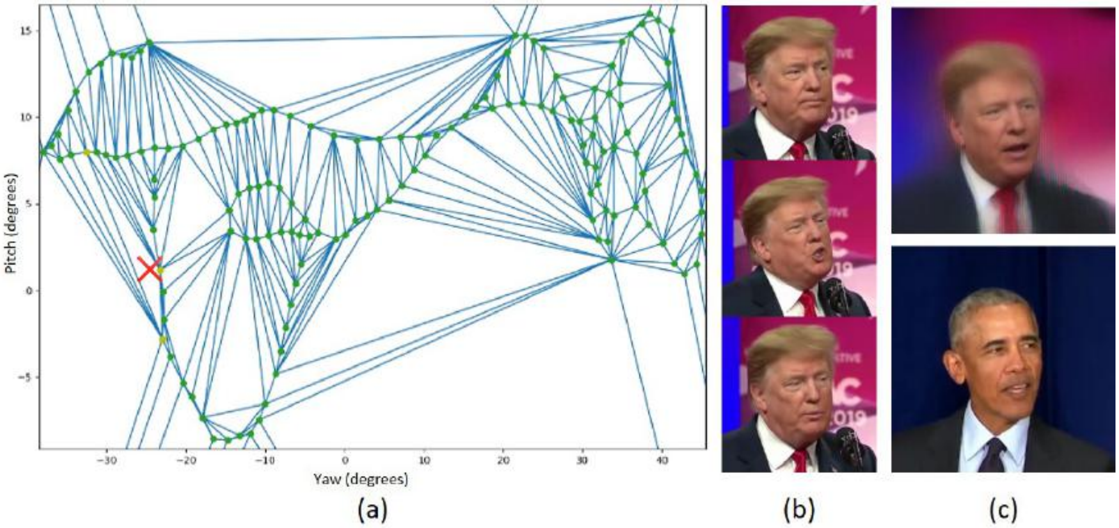

Figure 3: *Face view interpolation.* (a) Shows an example of an appearance map of the source subject (Donald Trump). The green dots represent different views of the source subject, the blue lines represent the Delaunay Triangulation of those views, and the red X marks the location of the current target's pose. (b) The interpolated views associated with the vertices of the selected triangle (represented by the yellow dots). (c) The reenactment result and the current target image.

spect to $x_{i_1}, x_{i_2}, x_{i_3}$. The interpolation result I_r is then

$$I_r = \sum_{k=1}^{3} \lambda_k G_r(I_{s_{i_k}}; H(\mathbf{p}_t)), \tag{9}$$

where $\mathbf{p}_t$ are the 2D landmarks of F_t. If any vertices of the triangle are boundary points, we exclude them from the interpolation and normalize the weights, λ_i, to sum to one.

A face view query is illustrated in Fig. 3(b,c). To improve interpolation accuracy, we use a horizontal flip to fill in views when the appearance map is one-sided with respect to the yaw dimension, and generate artificial views using G_r when the appearance map is too sparse.

3.4. Face inpainting

Occluded regions in the source face F_s cannot be rendered on the target face, F_t. Nirkin et al. [35] used the segmentations of F_s and F_t to remove occluded regions, rendering (swapping) only regions visible in both source and target faces. Large occlusions and different facial textures can cause noticeable artifacts in the resulting images.

To mitigate such problems, we apply a face inpainting generator, G_c (Fig. 2(b)). G_c renders face image F_s such that the resulting face rendering $\tilde{I}_r$ covers entire segmentation mask S_t (of F_t), thereby resolving such occlusion.

Given the reenactment result, I_r, its corresponding segmentation, S_r, and the target image with its background removed, $\tilde{I}_t$, all drawn from the same identity, we first augment S_r by simulating common face occlusions due to hair, by randomly removing ellipse-shaped parts, in various sizes and aspect ratios from the border of S_r. Let $\tilde{I}_r$ be I_r with its background removed using the augmented version of S_r, and I_c the completed result from applying G_c on $\tilde{I}_r$. We define our inpainting generator loss as

$$\mathcal{L}(G_c) = \lambda_{rec}\mathcal{L}_{rec}(I_c, \tilde{I}_t) + \lambda_{adv}\mathcal{L}_{adv}, \tag{10}$$

where $\mathcal{L}_{rec}$ and $\mathcal{L}_{adv}$ are the reconstruction and adversarial losses of Sec. 3.1.

3.5. Face blending

The last step of the proposed face swapping scheme is blending of the completed face F_c with its target face F_t (Fig. 2(c)). Any blending must account for, among others, different skin tones and lighting conditions. Inspired by previous uses of Poisson blending for inpainting [51] and blending [49], we propose a novel Poisson blending loss.

Let I_t be the target image, I_r^t the image of the reenacted face transferred onto the target image, and S_t the segmentation mask marking the transferred pixels. Following [36], we define the Poisson blending optimization as

$$\begin{aligned} P(I_t; I_r^t; S_t)) &= \arg\min_f \|\nabla f - \nabla I_r^t\|_2^2 \\ &\qquad \text{s.t. } f(i,j) = I_t(i,j),\ \forall\ S_t(i,j) = 0, \end{aligned} \tag{11}$$

where $\nabla(\cdot)$ is the gradient operator. We combine the Poisson optimization in Eq. (11) with the perceptual loss. The Poisson blending loss is then $\mathcal{L}(G_b)$

$$\mathcal{L}(G_b) = \lambda_{rec}\mathcal{L}_{rec}(G_b(I_t; I_r^t; S_t), P(I_t; I_r^t; S_t)) + \lambda_{adv}\mathcal{L}_{adv}.$$

4. Datasets and training

4.1. Datasets and processing

We use the video sequences of the IJB-C dataset [30] to train our generator, G_r, for which we automatically extracted the frames depicting particular subjects. IJB-C contains ∼11k face videos, of which we used 5,500 which were in high definition. Similar to the frame pruning approach of Sec. 3.3, we prune the face views that are too close together as well as motion-blurred frames.

We apply the segmentation CNN, G_s, to the frames, and prune the frames for which less than 15% of the pixels in the face bounding box were classified as face pixels. We used dlib's face verification[1] to group frames according to the subject identity, and limit the number of frames per subject to 100, by choosing frames with the maximal variance in 2D landmarks. In each training iteration, we choose the frames I_s and I_t from two randomly chosen subjects.

We trained VGG-19 CNNs for the perceptual loss on the VGGFace2 dataset [9] for face recognition and the CelebA [27] dataset for face attribute classification. The VGGFace2 dataset contains 3.3M images depicting 9,131 identities, whereas CelebA contains 202,599 images, annotated with 40 binary attributes.

We trained the segmentation CNN, G_s, on data used by others [35], consisting of $\sim 10k$ face images labeled with face segmentations. We also used the LFW Parts Labels set [19] with $\sim 3k$ images labeled for face and hair segmentations, removing the neck regions using facial landmarks.

We used additional 1k images and corresponding hair segmentations from the Figaro dataset [43]. Finally, FaceForensics++ [39] provides 1000 videos, from which they generated 1000 synthetic videos on random pairs using DeepFakes [12] and Face2Face [44].

4.2. Training details

We train the proposed generators from scratch, where the weights were initialized randomly using a normal distribution. We use Adam optimization [24] ($\beta_1 = 0.5, \beta_2 = 0.999$) and a learning rate of 0.0002. We reduce this rate by half every ten epochs. The following parameters were used for all the generators: $\lambda_{perc} = 1, \lambda_{pixel} = 0.1, \lambda_{adv} = 0.001, \lambda_{seg} = 0.1, \lambda_{rec} = 1, \lambda_{stepwise} = 1$, where $\lambda_{reenactment}$ is linearly increased from 0 to 1 during training. All of our networks were trained on eight NVIDIA Tesla V100 GPUs and an Intel Xeon CPU. Training of G_s required six hours to converge, while the rest of the networks converged in two days. All our networks, except for G_s, were trained using a progressive multi scale approach, starting with a resolution of 128×128 and ending at 256×256. Inference rate is ∼30fps for reenactment and ∼10fps for swapping on one NVIDIA Tesla V100 GPU.

5. Experimental results

We performed extensive qualitative and quantitative experiments to verify the proposed scheme. We compare our method to two previous face swapping methods: DeepFakes [12] and Nirkin et al. [35], and the Face2Face reenactment scheme [44]. We conduct all our experiments on videos from FaceForensics++ [39], by running our method on the same pairs they used. We further report ablation studies showing the importance of each component in our pipeline.

5.1. Qualitative face reenactment results

Fig. 4 shows our raw face reenactment results, without background removal. We chose examples of varying ethnicity, pose, and expression. A specifically interesting example can be seen in the rightmost column, showing our method's ability to cope with extreme expressions. To show the importance of iterative reenactment, Fig 5 provides reenactments of the same subject for both small and large angle differences. As evident from the last column, for large angle differences, the identity and texture are better preserved using multiple iterations.

5.2. Qualitative face swapping results

Fig. 6 offers face swapping examples taken from FaceForensics++ videos, *without training our model on these videos*. We chose examples that represent different poses and expression, face shapes, and hair occlusions. Because Nirkin et al. [35] is an image-to-image face swapping method, to be fair in our comparison, for each frame in the target video we select the source frame with the most similar pose. To compare FSGAN in a video-to-video scenario, we use our face view interpolation described in Sec. 3.3.

5.3. Comparison to Face2Face

We compare our method to Face2Face [44] on the expression only reenactment problem. Given a pair of faces $F_s \in I_s$ and $F_t \in I_t$ the goal is to transfer the expression from I_s to I_t. To this end, we modify the corresponding 2D landmarks of F_t by swapping in the mouth points of the 2D landmarks of F_s, similarly to how we generate the intermediate landmarks in Sec. 3.2. The reenactment result is then given by $G_r(I_t; H(\hat{p}_t))$, where $\hat{p}_t$ are the modified landmarks. The examples are shown in Fig. 7.

5.4. Quantitative results

We report quantitative results, conforming to how we defined the face swapping problem: we validate how well methods preserve the source subject identity, while retaining the same pose and expression of the target subject. To this end, we first compare the face swapping result, F_b, of each frame to its nearest neighbor in pose from the subject face views. We use the dlib [23] face verification method to compare identities and the structural similarity index method (SSIM) to compare their quality. To measure pose accuracy, we calculate the Euclidean distance between the

[1] Available: http://dlib.net/

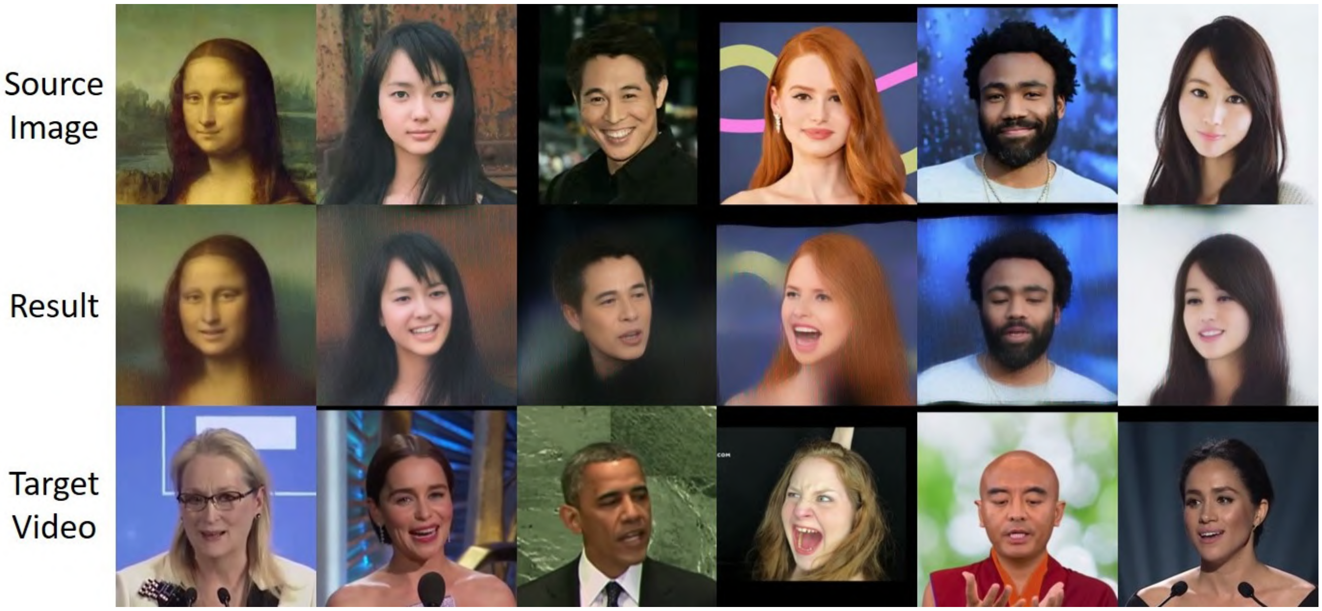

Figure 4: *Qualitative face reenactment results.* Row 1: The source face for reenactment. Row 2: Our reenactment results (without background removal). Row 3: The target face from which to transfer the pose and expression.

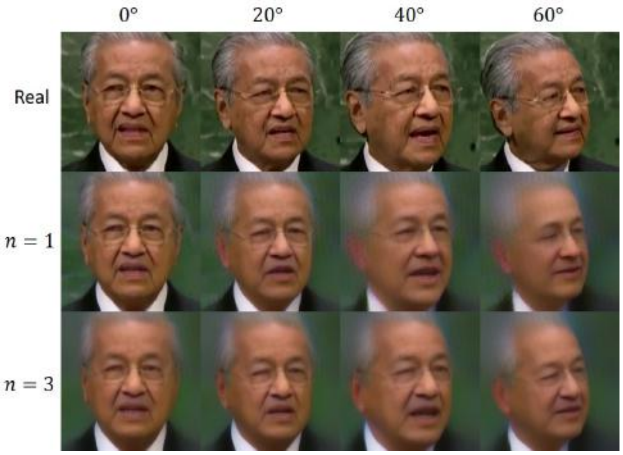

Figure 5: *Reenactment limitations.* Top left image transformed onto each of the images in Row 1 (using the same subject for clarity). Row 2: Reenactment with one iteration. Row 3: Three iterations.

Method	verification ↓	SSIM ↑	euler ↓	landmarks ↓
Nirkin et al. [35]	0.39 ± 0.00	0.49 ± 0.00	3.15 ± 0.04	26.5 ± 17.7
DeepFakes [12]	0.38 ± 0.00	0.50 ± 0.00	4.05 ± 0.04	34.1 ± 16.6
FSGAN	0.38 ± 0.00	**0.51 ± 0.00**	**2.49 ± 0.04**	**22.2 ± 17.7**

Table 1: *Quantitative swapping results.* On FaceForensics++ videos [39].

Euler angles of F_b to the original target image, I_t. Similarly, the accuracy of the expression is measured as the Euclidean distance between the 2D landmarks. Pose error is measured in degrees and the expression error is measured in pixels. We computes the mean and variance of those measurements on the first 100 frames of the first 500 videos in FaceForensics++, averaging them across the videos. As baselines, we use Nirkin et al. [35] and DeepFakes [12].

Evident from the first two columns of Table 1, our approach preserves identity and image quality similarly to previous methods. The two rightmost metrics in Table 1 show that our method retains pose and expression much better than its baselines. Note that the human eye is very sensitive to artifacts on faces. This should be reflected in the quality score but those artifacts usually capture only a small part of the image and so the SSIM score does not reflect them well.

5.5. Ablation study

We performed ablation tests with four configurations of our method: G_r only, $G_r + G_c$, $G_r + G_b$, and our full pipeline. The segmentation network, G_s, is used in all configurations. Qualitative results are provided in Fig. 8.

Quantitative ablation results are reported in Table 2. Verification scores show that source identities are preserved across all pipeline networks. From Euler and landmarks scores we see that target poses and expressions are best retained with the full pipeline. Error differences are not extreme, suggesting that the inpainting and blending generators, G_c and G_b, respectively, preserve pose and expression similarly well. There is a slight drop in the SSIM, due to the additional networks and processing added to the pipeline.

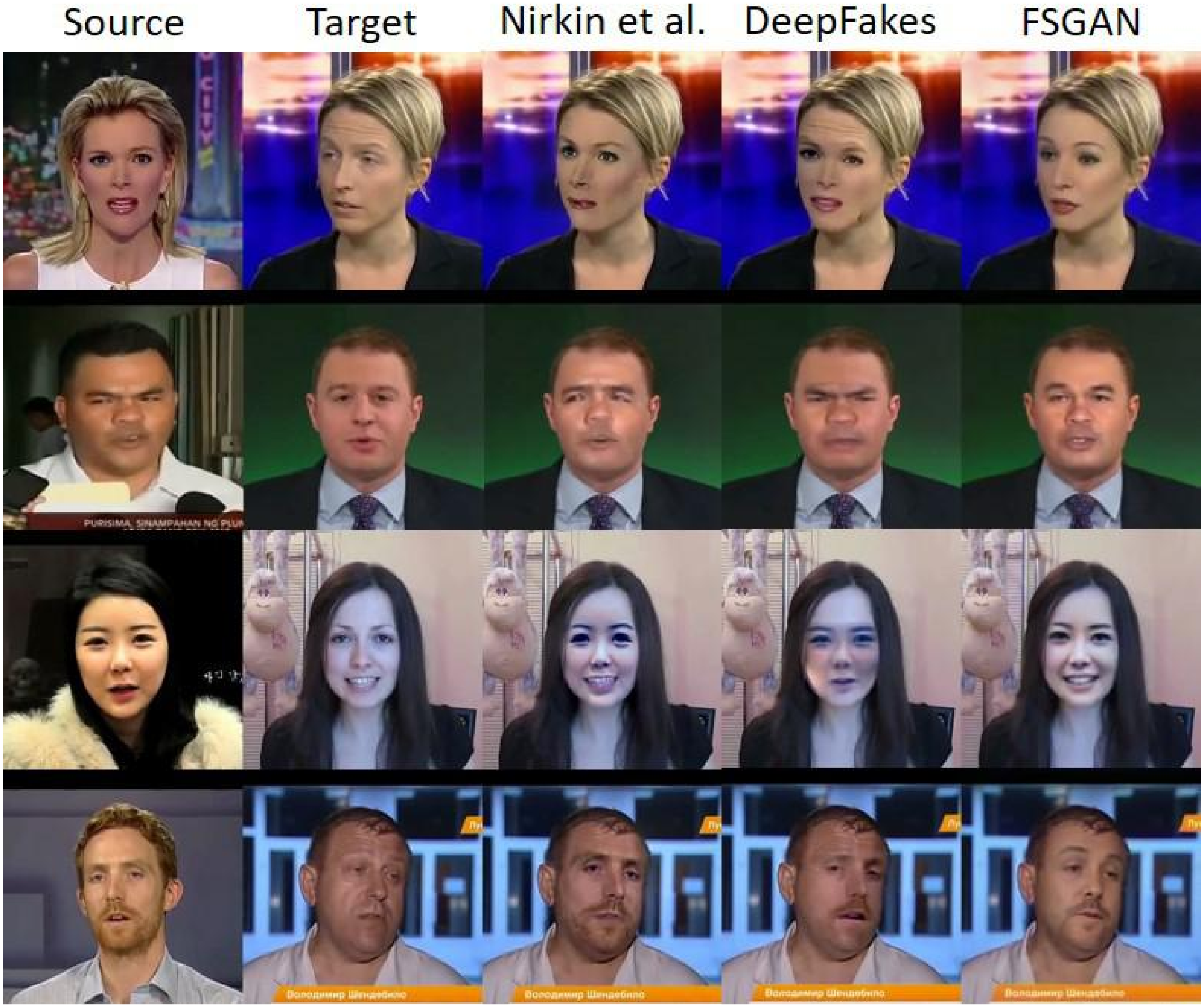

Figure 6: *Qualitative face swapping results on [39].* Results for source photo swapped onto target provided for Nirkin et al. [35], DeepFakes [12] and our method on images of faces of subjects it was not trained on.

Method	verification ↓	SSIM ↑	euler ↓	landmarks ↓
FSGAN (G_r)	0.38 ± 0.00	0.54 ± 0.00	3.16 ± 0.03	22.6 ± 16.5
FSGAN ($G_r + G_c$)	0.38 ± 0.00	0.54 ± 0.00	3.21 ± 0.08	24.5 ± 17.2
FSGAN ($G_r + G_b$)	0.38 ± 0.00	0.52 ± 0.00	2.75 ± 0.05	23.6 ± 17.9
FSGAN ($G_r + G_c + G_b$)	0.38 ± 0.00	0.51 ± 0.00	$\mathbf{2.49} \pm \mathbf{0.04}$	$\mathbf{22.2} \pm \mathbf{17.7}$

Table 2: *Quantitative ablation results.* On FaceForensics++ videos [39].

6. Conclusion

Limitations. Fig. 5 shows our reenactment results for different facial yaw angles. Evidently, the larger the angular differences, the more identity and texture quality degrade. Moreover, too many iterations of the face reenactment generator blur the texture. Unlike 3DMM based methods, e.g., Face2Face [44], which warp textures directly from the image, our method is limited to the resolution of the training data. Another limitation arises from using a sparse landmark tracking method that does not fully capture the complexity of facial expressions.

Discussion. Our method eliminates laborious, subject-specific, data collection and model training, making face swapping and reenactment accessible to non-experts. We feel strongly that it is of *paramount importance* to publish such technologies, in order to drive the development of technical counter-measures for detecting such forgeries, as well as compel law makers to set clear policies for addressing their implications. Suppressing the publication of such methods would not stop their development, but rather make them available to select few and potentially blindside

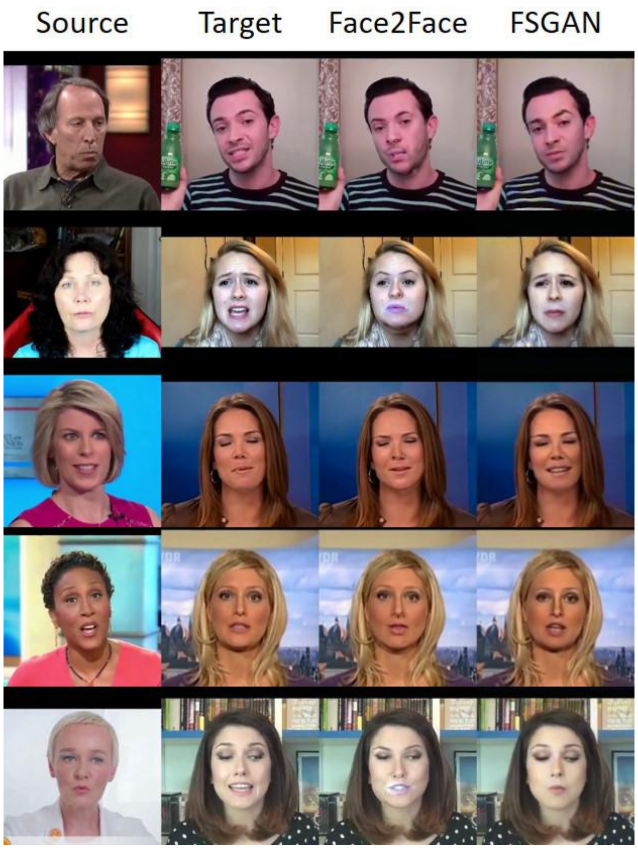

Figure 7: *Comparison to Face2Face [44] on FaceForensics++ [39].* As demonstrated, our method exhibits far less artifacts than Face2Face.

policy makers if it is misused.

References

[1] Oleg Alexander, Mike Rogers, William Lambeth, Matt Chiang, and Paul Debevec. Creating a photoreal digital actor: The digital emily project. In *Conf. Visual Media Production*, pages 176–187. IEEE, 2009.

[2] Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein gan. *arXiv preprint arXiv:1701.07875*, 2017.

[3] Hadar Averbuch-Elor, Daniel Cohen-Or, Johannes Kopf, and Michael F Cohen. Bringing portraits to life. *ACM Transactions on Graphics (TOG)*, 36(6):196, 2017.

[4] Dmitri Bitouk, Neeraj Kumar, Samreen Dhillon, Peter Belhumeur, and Shree K Nayar. Face swapping: automatically replacing faces in photographs. *ACM Trans. on Graphics*, 27(3):39, 2008.

[5] Volker Blanz, Sami Romdhani, and Thomas Vetter. Face identification across different poses and illuminations with a 3d morphable model. In *Int. Conf. on Automatic Face and Gesture Recognition*, pages 192–197, 2002.

[6] Volker Blanz, Kristina Scherbaum, Thomas Vetter, and Hans-Peter Seidel. Exchanging faces in images. *Comput. Graphics Forum*, 23(3):669–676, 2004.

[7] Volker Blanz and Thomas Vetter. Face recognition based on fitting a 3d morphable model. *Trans. Pattern Anal. Mach. Intell.*, 25(9):1063–1074, 2003.

[8] Xavier P Burgos-Artizzu, Pietro Perona, and Piotr Dollár. Robust face landmark estimation under occlusion. In *Proc. Int. Conf. Comput. Vision*, pages 1513–1520. IEEE, 2013.

[9] Qiong Cao, Li Shen, Weidi Xie, Omkar M Parkhi, and Andrew Zisserman. Vggface2: A dataset for recognising faces across pose and age. In *2018 13th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2018)*, pages 67–74. IEEE, 2018.

[10] Feng-Ju Chang, Anh Tran, Tal Hassner, Iacopo Masi, Ram Nevatia, and Gérard Medioni. FacePoseNet: Making a case for landmark-free face alignment. In *Proc. Int. Conf. Comput. Vision Workshops*, 2017.

[11] Feng-Ju Chang, Anh Tuan Tran, Tal Hassner, Iacopo Masi, Ram Nevatia, and Gérard Medioni. Deep, landmark-free fame: Face alignment, modeling, and expression estimation. *Int. J. Comput. Vision*, 127(6-7):930–956, 2019.

[12] DeepFakes. FaceSwap. `https://github.com/deepfakes/faceswap`. Accessed: 2019-02-06.

[13] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014.

[14] Tal Hassner, Shai Harel, Eran Paz, and Roee Enbar. Effective face frontalization in unconstrained images. In *Proc. Conf. Comput. Vision Pattern Recognition*, 2015.

[15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[16] John F Hughes, Andries Van Dam, James D Foley, and Steven K Feiner. *Computer graphics: principles and practice*. Pearson Education, 2014.

[17] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1125–1134, 2017.

[18] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *European Conference on Computer Vision*, pages 694–711. Springer, 2016.

[19] Andrew Kae, Kihyuk Sohn, Honglak Lee, and Erik Learned-Miller. Augmenting crfs with boltzmann machine shape priors for image labeling. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2019–2026, 2013.

[20] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of gans for improved quality, stability, and variation. *arXiv preprint arXiv:1710.10196*, 2017.

[21] Ira Kemelmacher-Shlizerman. Transfiguring portraits. *ACM Trans. on Graphics*, 35(4):94, 2016.

[22] Hyeongwoo Kim, Pablo Carrido, Ayush Tewari, Weipeng Xu, Justus Thies, Matthias Niessner, Patrick Pérez, Christian Richardt, Michael Zollhöfer, and Christian Theobalt.

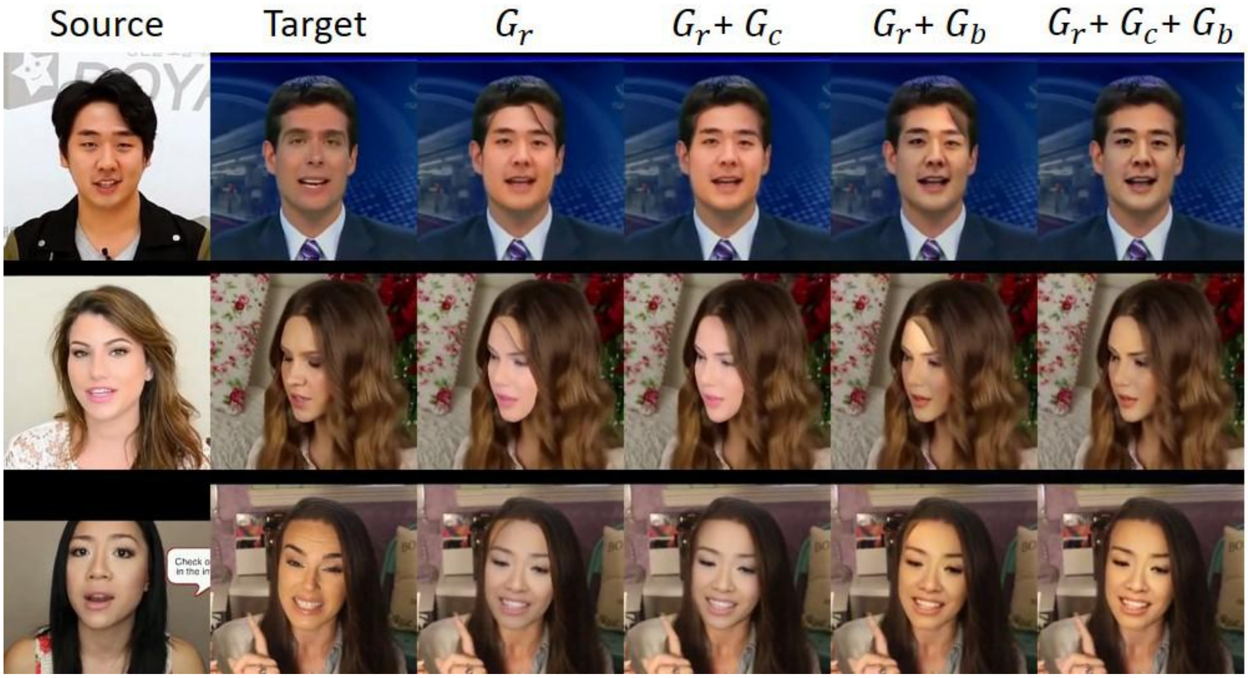

Figure 8: *Ablation study*. From columns 3 and 5, without the completion network, G_c, the transferred face does not cover the entire target face, leaving obvious artifacts. Columns 3 and 4 show that without the blending network, G_b, the skin color and lighting conditions of the transferred face are inconsistent with its new context.

Deep video portraits. *ACM Transactions on Graphics (TOG)*, 37(4):163, 2018.

[23] Davis E. King. Dlib-ml: A machine learning toolkit. *Journal of Machine Learning Research*, 10:1755–1758, 2009.

[24] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[25] Christian Ledig, Lucas Theis, Ferenc Huszár, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, et al. Photo-realistic single image super-resolution using a generative adversarial network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4681–4690, 2017.

[26] Yuan Lin, Shengjin Wang, Qian Lin, and Feng Tang. Face swapping under large pose variations: A 3D model based approach. In *Int. Conf. on Multimedia and Expo*, pages 333–338. IEEE, 2012.

[27] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Large-scale celebfaces attributes (celeba) dataset. *Retrieved August*, 15:2018, 2018.

[28] Xudong Mao, Qing Li, Haoran Xie, Raymond YK Lau, Zhen Wang, and Stephen Paul Smolley. Least squares generative adversarial networks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2794–2802, 2017.

[29] Iacopo Masi, Anh Tun Trn, Tal Hassner, Gozde Sahin, and Gérard Medioni. Face-specific data augmentation for unconstrained face recognition. *Int. J. Comput. Vision*, 127(6-7):642–667, 2019.

[30] Brianna Maze, Jocelyn Adams, James A Duncan, Nathan Kalka, Tim Miller, Charles Otto, Anil K Jain, W Tyler Niggel, Janet Anderson, Jordan Cheney, et al. Iarpa janus benchmark-c: Face dataset and protocol. In *2018 International Conference on Biometrics (ICB)*, pages 158–165. IEEE, 2018.

[31] Mehdi Mirza and Simon Osindero. Conditional generative adversarial nets. *arXiv preprint arXiv:1411.1784*, 2014.

[32] Saleh Mosaddegh, Loic Simon, and Frédéric Jurie. Photorealistic face de-identification by aggregating donors face components. In *Asian Conf. Comput. Vision*, pages 159–174. Springer, 2014.

[33] Ryota Natsume, Tatsuya Yatagawa, and Shigeo Morishima. Fsnet: An identity-aware generative model for image-based face swapping. In *Proc. of Asian Conference on Computer Vision (ACCV)*. Springer, Dec 2018.

[34] Ryota Natsume, Tatsuya Yatagawa, and Shigeo Morishima. Rsgan: face swapping and editing using face and hair representation in latent spaces. *arXiv preprint arXiv:1804.03447*, 2018.

[35] Yuval Nirkin, Iacopo Masi, Anh Tran Tuan, Tal Hassner, and Gerard Medioni. On face segmentation, face swapping, and face perception. In *Automatic Face & Gesture Recognition (FG 2018), 2018 13th IEEE International Conference on*, pages 98–105. IEEE, 2018.

[36] Patrick Pérez, Michel Gangnet, and Andrew Blake. Poisson image editing. *ACM Transactions on graphics (TOG)*, 22(3):313–318, 2003.

[37] Albert Pumarola, Antonio Agudo, Aleix M Martinez, Alberto Sanfeliu, and Francesc Moreno-Noguer. Ganimation:

Anatomically-aware facial animation from a single image. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 818–833, 2018.

[38] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.

[39] Andreas Rössler, Davide Cozzolino, Luisa Verdoliva, Christian Riess, Justus Thies, and Matthias Nießner. Faceforensics++: Learning to detect manipulated facial images. *arXiv*, 2019.

[40] Enrique Sanchez and Michel Valstar. Triple consistency loss for pairing distributions in gan-based face synthesis. *arXiv preprint arXiv:1811.03492*, 2018.

[41] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[42] Supasorn Suwajanakorn, Steven M Seitz, and Ira Kemelmacher-Shlizerman. Synthesizing obama: learning lip sync from audio. *ACM Transactions on Graphics (TOG)*, 36(4):95, 2017.

[43] Michele Svanera, Umar Riaz Muhammad, Riccardo Leonardi, and Sergio Benini. Figaro, hair detection and segmentation in the wild. In *2016 IEEE International Conference on Image Processing (ICIP)*, pages 933–937. IEEE, 2016.

[44] Justus Thies, Michael Zollhofer, Marc Stamminger, Christian Theobalt, and Matthias Nießner. Face2face: Real-time face capture and reenactment of rgb videos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2387–2395, 2016.

[45] Yu Tian, Xi Peng, Long Zhao, Shaoting Zhang, and Dimitris N Metaxas. Cr-gan: learning complete representations for multi-view generation. *arXiv preprint arXiv:1806.11191*, 2018.

[46] Anh Tuan Tran, Tal Hassner, Iacopo Masi, Eran Paz, Yuval Nirkin, and Gérard Medioni. Extreme 3D face reconstruction: Looking past occlusions. In *Proc. Conf. Comput. Vision Pattern Recognition*, 2018.

[47] Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. High-resolution image synthesis and semantic manipulation with conditional gans. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[48] Lior Wolf, Ziv Freund, and Shai Avidan. An eye for an eye: A single camera gaze-replacement method. In *Proc. Conf. Comput. Vision Pattern Recognition*, pages 817–824. IEEE, 2010.

[49] Huikai Wu, Shuai Zheng, Junge Zhang, and Kaiqi Huang. Gp-gan: Towards realistic high-resolution image blending. *arXiv preprint arXiv:1703.07195*, 2017.

[50] Wayne Wu, Yunxuan Zhang, Cheng Li, Chen Qian, and Chen Change Loy. Reenactgan: Learning to reenact faces via boundary transfer. In *ECCV*, Sept. 2018.

[51] Raymond A Yeh, Chen Chen, Teck Yian Lim, Alexander G Schwing, Mark Hasegawa-Johnson, and Minh N Do. Semantic image inpainting with deep generative models. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5485–5493, 2017.

[52] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2223–2232, 2017.

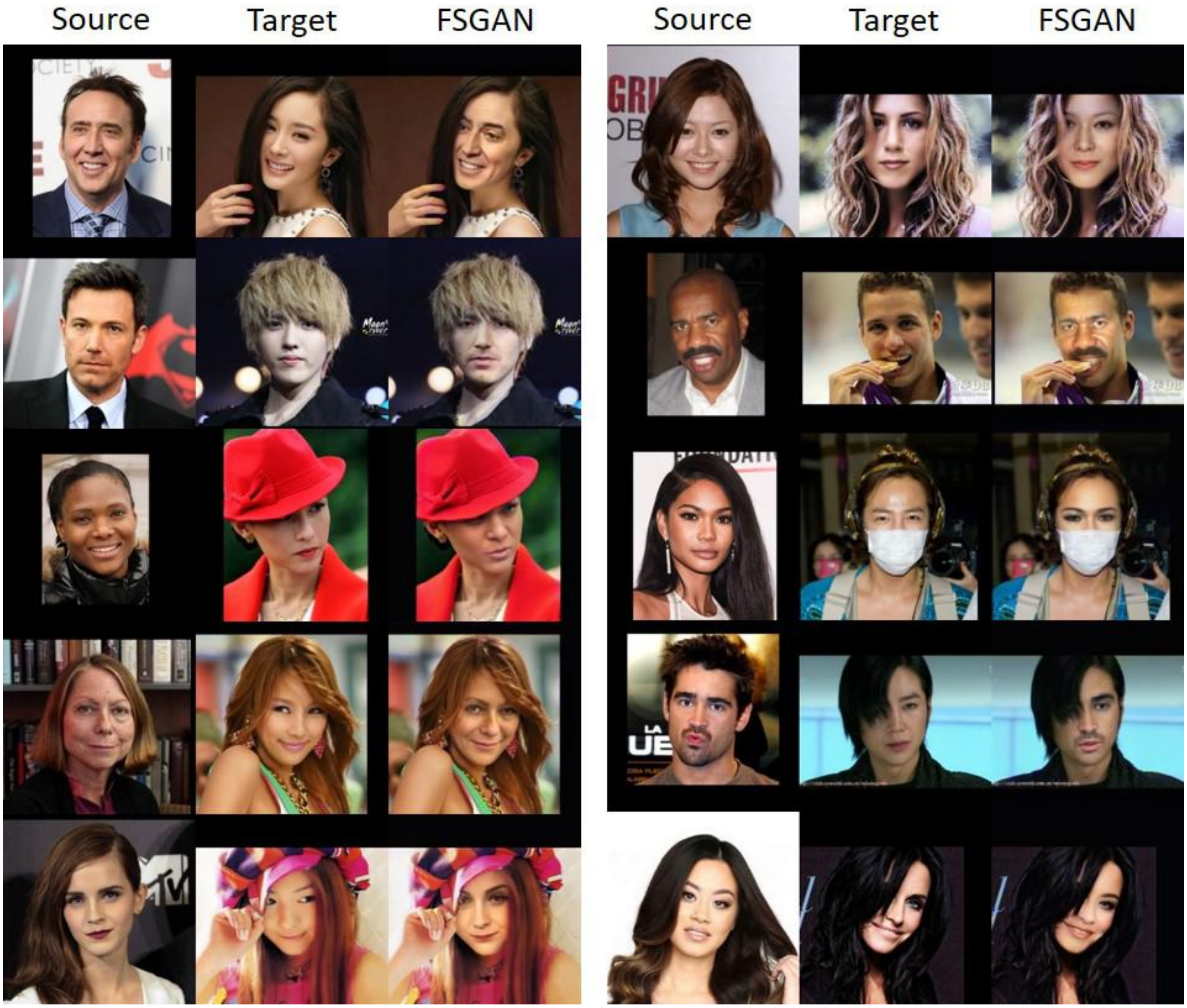

Figure 9: *Additional qualitative face swapping results on on the Caltech Occluded Faces in the Wild (COFW) dataset [8].*

Supplementary Material

A. Additional qualitative results

We offer additional quantitative face swapping results in Fig. 9. We have specifically chosen examples of challenging pairs, with partial occlusions, different ethnicities and skin colors, demonstrating the competence of our method on a large variety of subjects. In Fig. 10, we show additional quantitative comparison to Nirkin et al. [35] and DeepFakes [12], and in Fig. 11 we show another comparison to Face2Face [44]. Please also see the attached video for more results.

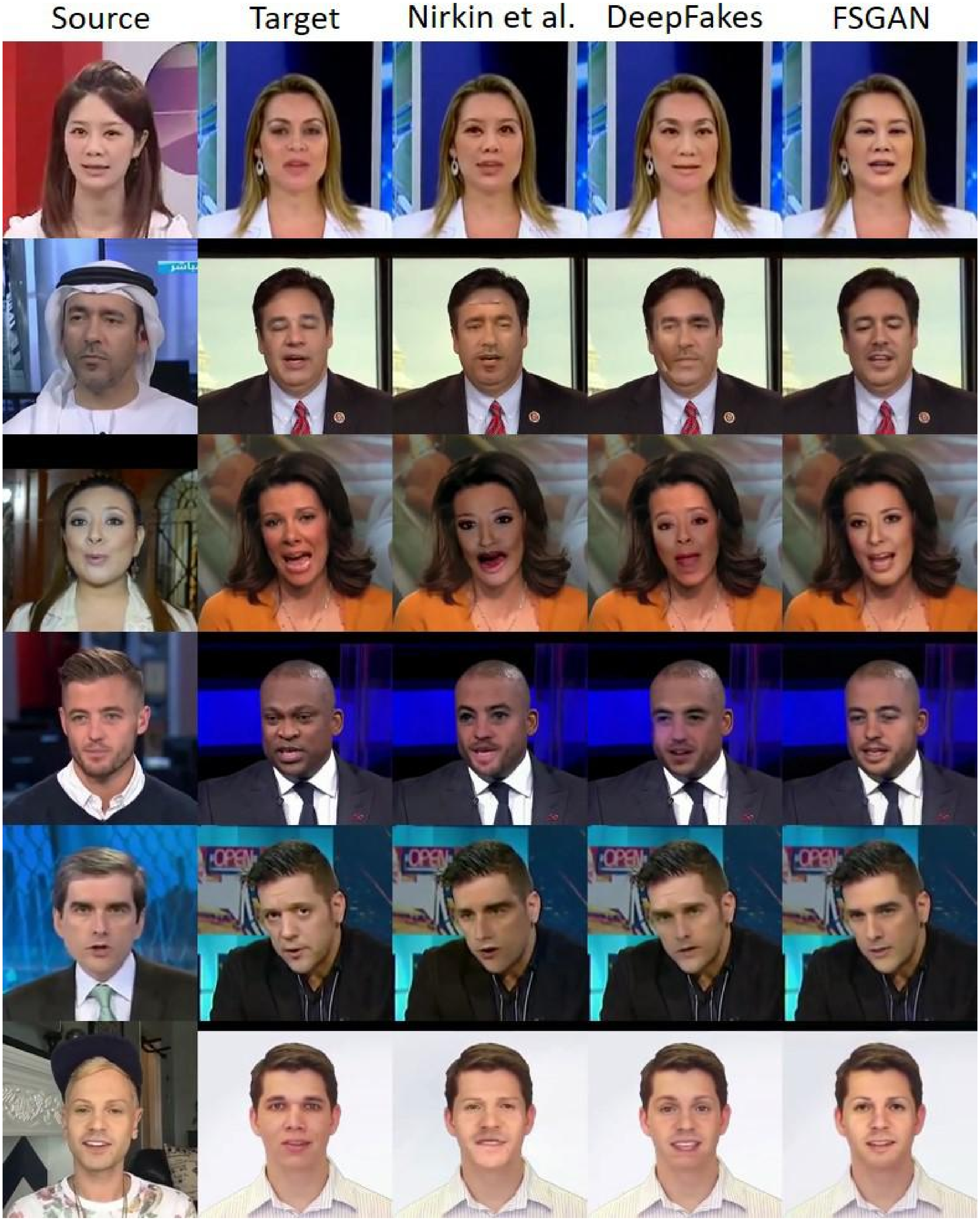

Figure 10: *Additional qualitative face swapping comparison to Nirkin et al. [35] and DeepFakes [12] on FaceForensics++ [39].*

B. The architecture of the generator CNNs

The architecture of the generators, G_r, G_c, and G_b, is based on the pix2pixHD approach [47], and the layout of the global generator and enhancer is depicted in Fig. 12. The global generator is defined by the number of bottleneck blocks (shown in purple) used in each resolution scale. In our experiments we used only three resolutions. The enhancer is defined by its submodule, that is, either the global generator or another enhancer, and its number of bottleneck layers. The generators are thus given by

$$G_r = G_c = \text{Enhancer}(\text{Global}(2, 2, 3), 2),$$

and

$$G_b = \text{Enhancer}(\text{Global}(1, 1, 1), 1).$$

The face segmentation network G_s is based on the U-Net approach [38], for which we replaced the deconvolution layers with bilinear interpolation upsampling layers.

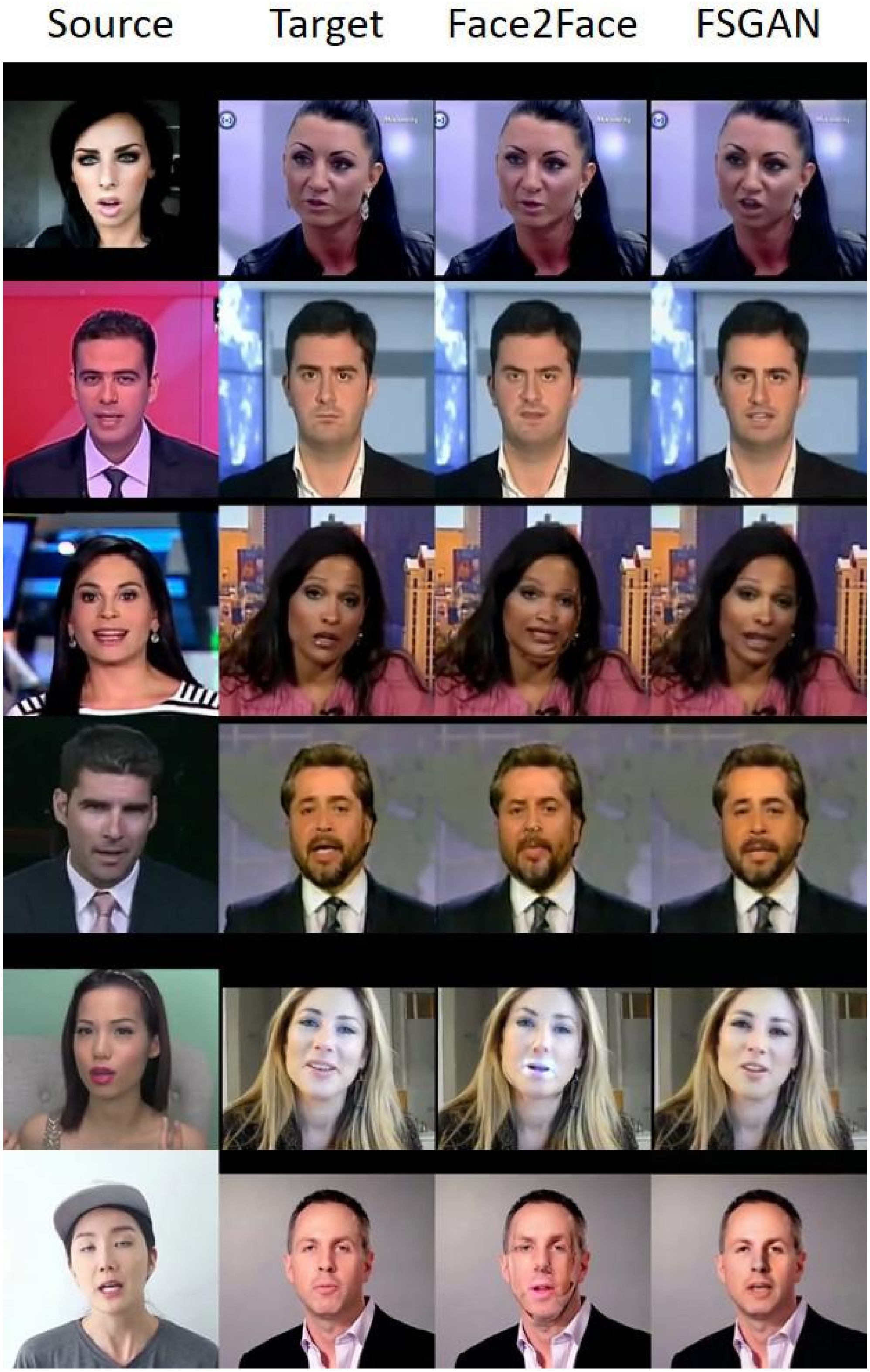

Figure 11: *Additional qualitative face reenactment comparison to Face2Face [44] on FaceForensics++ [39].*

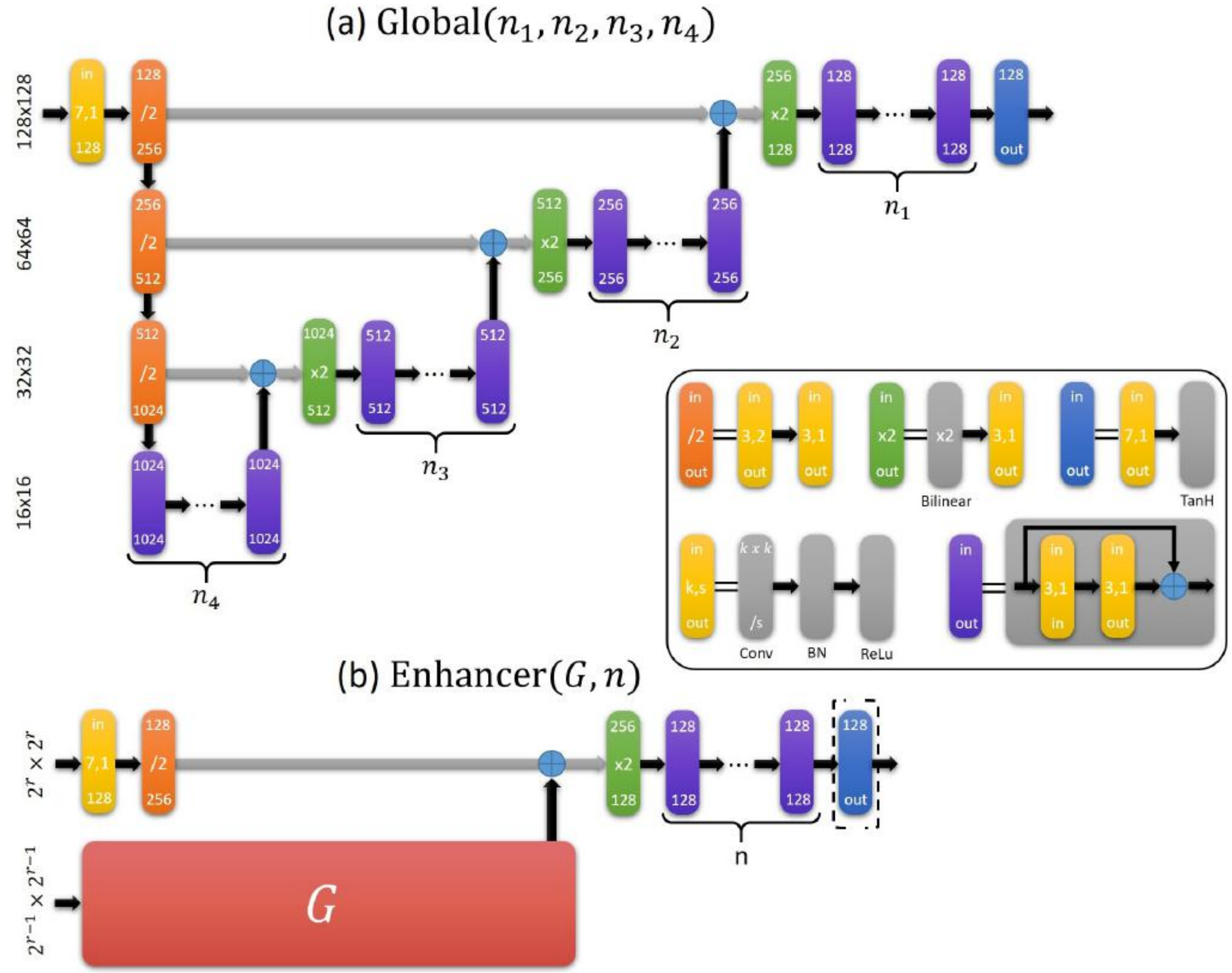

Figure 12: *Generator architectures.* (a) The global generator is based on a residual variant of the U-Net [38] CNN, using a number of bottleneck layers per resolution. We replace the simple convolutions with bottleneck blocks (in purple), the concatenation with summation (plus sign), and the deconvolutions with bilinear upsamplnig following by a convolution. (b) The enhancer utilizes a submodule and a number of bottleneck layers. The last output block (in blue) is only used in the enhancer of the finest resolution.