Advancing High Fidelity Identity Swapping for Forgery Detection

Lingzhi Li1* Jianmin Bao2† Hao Yang2 Dong Chen2 Fang Wen2
1Peking University 2Microsoft Research
lilingzhi@pku.edu.cn {jianbao,haya,doch,fangwen}@microsoft.com

Abstract

In this work, we study various existing benchmarks for deepfake detection researches. In particular, we examine a novel two-stage face swapping algorithm, called FaceShifter, for high fidelity and occlusion aware face swapping. Unlike many existing face swapping works that leverage only limited information from the target image when synthesizing the swapped face, FaceShifter generates the swapped face with high-fidelity by exploiting and integrating the target attributes thoroughly and adaptively. FaceShifter can handle facial occlusions with a second synthesis stage consisting of a Heuristic Error Acknowledging Refinement Network (HEAR-Net), which is trained to recover anomaly regions in a self-supervised way without any manual annotations. Experiments show that existing deepfake detection algorithm performs poorly with FaceShifter, since it achieves advantageous quality over all existing benchmarks. However, our newly developed Face X-Ray [23] method can reliably detect forged images created by FaceShifter.

1. Introduction

Face swapping has attracted great interest in the vision and graphics community, because of its potential wide applications in movie composition, computer games, and privacy protection [35]. More worth mentioning is that better face swapping technologies will help build better face forgery detection technologies.

Recent research [36] shows that the previous face swapping algorithms can be easily detected by a binary classifier. This is because the quality of synthetic faces from these algorithms are usually unsatisfactory. Early replacement-based works [6, 42] simply replace the pixels of inner face region. Thus, they are sensitive to the variations in posture and perspective. 3D-based works [7, 12, 26, 31] used a 3D model to deal with the posture or perspective difference. However the accuracy and robustness of 3D reconstruction of faces are all unsatisfactory. Recently, GAN-based works [21, 28, 29, 30, 4] have illustrated impressive results. But it remains challenging to synthesize both realistic and high-fidelity results.

In this work, we focus on improving the fidelity of face swapping, and examine the face forgery face detection algorithm on the this new face swapping algorithm. In order to make the results more perceptually appealing, it is important that the synthesized swapped face not only shares the pose and expression of the target face, but also can be seamlessly fitted into the target image without inconsistency: the rendering of the swapped face should be faithful to the lighting (e.g. direction, intensity, color) of the target scene, the pixel resolution of the swapped face should also be consistent with the target image resolution. Neither of these can be well handled by a simple alpha or Poisson blending. Instead, we need a thorough and adaptive integration of target image attributes during the synthesis of the swapped face, so that the attributes from the target image, including scene lighting or image resolution, can help make the swapped face more realistic.

However, previous face swapping methods either neglect the requirement of this integration, or lack the ability to perform it in a thorough and adaptive way. In specific, many previous methods use only pose and expression guidances from the target image to synthesize the swapped face, the face is then blended into the target image using masks of
While our method addresses all these issues. Moreover, handling facial occlusions is always challenging in face swapping. Unlike Nirkin et al. [30,31] that trains face segmentation to obtain occlusion-aware face masks, our method can learn to recover face anomaly regions in a self-supervised way without any manual annotations. We observe that when feeding the same face image as both the target and source into a well trained AEI-Net, the reconstructed face image deviates from the input in multiple areas, these deviations strongly hint the locations of face occlusions. Thus, we propose a novel Heuristic Error Acknowledging Refinement Network (HEAR-Net) to further refine the result under the guidance of such reconstruction errors. The proposed method is more general, thus it identifies more anomaly types, such as glasses, shadow and reflection effects, and other uncommon occlusions.

The proposed two-stage face swapping framework, FaceShifter, is subject agnostic. Once trained, the model can be applied to any new face pairs without requiring subject specific training as DeepFakes [1] and Korshunova et al. [21]. Experiments demonstrate that our method achieves results considerably more realistic and more faithful to inputs than other state-of-the-art methods.

2. Related Works

Face swapping has a long history in vision and graphics researches. Early efforts [6,42] only swap faces with similar poses. Such a limitation is addressed by recent algorithms roughly divided in two categories: 3D-based approaches and GAN-based approaches.

3D-Based Approaches. Blanz et al. [7] considers 3D transform between two faces with different poses, but requiring user interaction and not handling expressions. Thies et al. [39] captures head actions from a RGB-D image using 3DMM, turning a static face into a controllable avatar. It is extended for RGB references in Face2Face [40]. Olszewski et al. [32] dynamically infereces 3D face textures for improved manipulation quality. Kim et al. [20] separately models different videos using 3DMM to make the portraits controllable, while Nagano et al. [27] needs only one image to reenact the portrait within. Recently, Thies et al. [38] adopt neural textures, which can better disentanglement geometry in face reenactment. However, when applied on face swapping, these methods hardly leverage target attributes like occlusions, lighting or photo styles. To preserve the target facial occlusions, Nirkin et al. [31,30] collected data to train an occlusion-aware face segmentation network in a supervised way, which helps predict a visible target face mask for blending in the swapped face. While our method find the occlusions in a self-supervised way without any manually annotations.

GAN-Based Approaches. In the GAN-based face swapping methods, Korshunova et al. [22] swap faces like transfer styles. It separately models different source identities, such as a CageNet for Nicolas Cage, a SwiftNet for Taylor Swift. The recently popular DeepFakes [1] is another example of such subject-aware face swapping: for each new in-
put, a new model has to be trained on two video sequences, one for the source and one for the target.

This limitation has been addressed by subject-agnostic face swapping researches: RSGAN [29] learns to extract vectorized embeddings for face and hair regions separately, and recombines them to synthesize a swapped face. FSNet [28] represents the face region of source image as a vector, which is combined with a non-face target image to generate the swapped face. IPGAN [5] disentangles the identity and attributes of faces as vectors. By introducing supervisions directly from the source identity and the target image, IPGAN supports face swapping with better identity preservation. However, due to the information loss caused by the compressed representation, and the lack of more adaptive information integration, these three methods are incapable of generating high-quality face images. Recently, FSGAN [30] performs face reenactment and face swapping together. It follows a similar reenact and blend strategy with [32, 27]. Although FSGAN utilizes an occlusion-aware face segmentation network for preserving target occlusions, it hardly respects target attributes like the lighting or image resolution, it can neither preserve the face shape of the source identity.

3. Methods

Our method requires two input images, i.e., a source image $X_s$ to provide identity and a target image $X_t$ to provide attributes, e.g., pose, expression, scene lighting and background. The swapped face image is generated through a two-stage framework, called FaceShifter. In the first stage, we use an Adaptive Embedding Integration Network (AEI-Net) to generate a high fidelity face swapping result $\hat{Y}_{s,t}$ based on information integration. In the second stage, we use the Heuristic Error Acknowledging Network (HEAR-Net) to handle the facial occlusions and refine the result, the final result is denoted by $Y_{s,t}$.

3.1. Adaptive Embedding Integration Network

In the first stage, we aim to generate a high fidelity face image $\hat{Y}_{s,t}$, which should preserve the identity of the source $X_s$ and the attributes (e.g., pose, expression, lighting, background) of the target $X_t$. To achieve this goal, our method consists of 3 components: i) the Identity Encoder $z_{id}(X_s)$, which extracts identity from the source image $X_s$; ii) the Multi-level Attributes Encoder $z_{att}(X_t)$, which extracts attributes of the target image $X_t$; iii) Adaptive Attentional Denormalization (AAD) Generator, which generates swapped face image. Figure 3(a) shows whole network structure.

**Identity Encoder**: We use a pretrained state-of-the-art face recognition model [13] as identity encoder. The identity embedding $z_{id}(X_s)$ is defined to be the last feature vector generated before the final FC layer. We believe that by training on a large quantity of 2D face data, such a face recognition model can provide more representative identity embeddings than the 3D-based models like 3DMM [7, 8].

**Multi-level Attributes Encoder**: Face attributes, such as pose, expression, lighting and background, require more spatial informations than identity. In order to preserve such details, we propose to represent the attributes embedding as multi-level feature maps, instead of compressing it into a single vector as previous methods [5, 29]. In specific, we feed the target image $X_t$ into a U-Net-like structure. Then we define the attributes embedding as the feature maps generated from the U-Net decoder. More formally, we define

$$z_{att}(X_t) = \{z_{att}^1(X_t), z_{att}^2(X_t), \ldots, z_{att}^n(X_t)\},$$

where $z_{att}^k(X_t)$ represents the $k$-th level feature map from the U-Net decoder, $n$ is the number of feature levels.

Our attributes embedding network does not require any attribute annotations, it extracts the attributes using self-supervised training: we require that the generated swapped face $Y_{s,t}$ and the target image $X_t$ have the same attributes embedding. The loss function will be introduce in Equation 7. In the experimental part (Section 4.2), we also study what the attributes embedding has learned.

**Adaptive Attentional Denormalization Generator**: We then integrate such two embeddings $z_{id}(X_s)$ and $z_{att}(X_t)$ for generating a raw swapped face $Y_{s,t}$. Previous methods [5, 29] simply integrate them through feature concatenation. It will lead to relatively blurry results. Instead, we propose a novel Adaptive Attentional Denormalization (AAD) layer to accomplish this task in a more adaptive fashion. Inspired by the mechanisms of SPADE [33] and AdaIN [14, 16], the proposed AAD layers leverage denormalizations for feature integration in multiple feature levels.

As shown in Figure 3(c), in the $h$-th feature level, let $h_{in}^k$ denote the activation map that is fed into an AAD layer, which should be a 3D tensor of size $C_h \times H_h \times W_h$, with $C_h$ being the number of channels and $H_h \times W_h$ being the spatial dimensions. Before integration, we perform batch normalization [17] on $h_{in}^k$:

$$\bar{h}^k = \frac{h_{in}^k - \mu^k}{\sigma^k}.$$  \hspace{1cm} (2)

Here $\mu^k \in \mathbb{R}^{C_h}$ and $\sigma^k \in \mathbb{R}^{C_h}$ are the means and standard deviations of the channel-wise activations within $h_{in}^k$ mini-batch. Then, we design 3 parallel branches for $\bar{h}^k$ for 1) attributes integration, 2) identity integration, 3) adaptively attention mask.

For attributes embedding integration, let $z^k_{att}$ be the attributes embedding on this feature level, which should be a 3D tensor of size $C_{att} \times H_h \times W_h$. In order to integrate $z^k_{att}$ into the activation, we compute an attribute activation $A^k$ by denormalizing the normalized $\bar{h}^k$ according to the attributes embedding, formulated as

$$A^k = \gamma^k_{att} \odot \bar{h}^k + \beta^k_{att},$$ \hspace{1cm} (3)
where $\gamma_{att}^k$ and $\beta_{att}^k$ are two modulation parameters both convolved from $z_{id}$. They share the same tensor dimensions with $h^k$. The computed $\gamma_{att}^k$ and $\beta_{att}^k$ are multiplied and added to $h^k$ element-wise.

For identity embedding integration, let $z_{id}^k$ be the identity embedding, which should be a 1D vector of size $C_{id}$. We also integrate $z_{id}^k$ by computing an identity activation $I^k$ in a similar way to integrating attributes. It is formulated as

$$I^k = \gamma_{id}^k \otimes h^k + \beta_{id}^k,$$

where $\gamma_{id}^k \in \mathbb{R}^{C_{id}}$ and $\beta_{id}^k \in \mathbb{R}^{C_{id}}$ are another two modulation parameters generated from $z_{id}$ through FC layers.

One key design of the AAD layer is to adaptively adjust the effective regions of the identity embedding and the attributes embedding, so that they can participate in synthesizing different parts of the face. For example, the identity embedding should focus relatively more on synthesizing the face parts that are most discriminative for distinguishing identities, e.g., eyes, mouth and face contour. Therefore, we adopt an attention mechanism into the AAD layer. Specifically, we generate an attentional mask $M^k$ using $h^k$ through convolutions and a sigmoid operation. The values of $M^k$ are between 0 and 1.

Finally, the output of this AAD layer $h_{out}^k$ can be obtained as a element-wise combination of the two activations $A^k$ and $I^k$, weighted by the mask $M^k$, as shown in Figure 3(c). It is formulated as

$$h_{out}^k = (1 - M^k) \otimes A^k + M^k \otimes I^k.$$  (5)

The AAD-Generator is then built with multiple AAD layers. As illustrated in Figure 3(a), after extracting the identity embedding $z_{id}$ from source $X_s$, and the attributes embedding $z_{att}$ from target $X_t$, we cascade AAD Residual Blocks (AAD ResBlks) to generate the swapped face $\hat{Y}_{s,t}$. The structure of the AAD ResBlks is shown in Figure 3(b). For the AAD ResBlk on the $k$-th feature level, it first takes the up-sampled activation from the previous level as input, then integrates this input with $z_{id}$ and $z_{att}$. The final output image $\hat{Y}_{s,t}$ is convolved from the last activation.

### Training Losses

We utilize adversarial training for AEI-Net. Let $\mathcal{L}_{adv}$ be the adversarial loss for making $\hat{Y}_{s,t}$ realistic. It is implemented as a multi-scale discriminator [33] on the downsampled output images. In addition, an identity preservation loss is used to preserve the identity of the source. It is formulated as

$$\mathcal{L}_{id} = 1 - \cos(z_{id}(\hat{Y}_{s,t}), z_{id}(X_s)),$$  (6)

where $\cos(\cdot, \cdot)$ represents the cosine similarity of two vectors. We also define the attributes preservation loss as $\mathcal{L}_{att}$, which is a 2 distances between the multi-level attributes embeddings from $X_t$ and $Y_{s,t}$. It is formulated as

$$\mathcal{L}_{att} = \frac{1}{2} \sum_{k=1}^{n} \|z_{att}^k(\hat{Y}_{s,t}) - z_{att}^k(X_t)\|_2^2.$$  (7)

When the source and target images are the same in a training sample, we define a reconstruction loss as pixel level $\mathcal{L}_{rec}$, which is a 2 distances between the target image $X_t$ and $Y_{s,t}$.

$$\mathcal{L}_{rec} = \begin{cases} \frac{1}{2} \|\hat{Y}_{s,t} - X_s\|_2^2 & \text{if } X_t = X_s \\ 0 & \text{otherwise} \end{cases}.$$  (8)

The AEI-Net is finally trained with a weighted sum of above losses as

$$\mathcal{L}_{AEI-Net} = \mathcal{L}_{adv} + \lambda_{att} \mathcal{L}_{att} + \lambda_{id} \mathcal{L}_{id} + \lambda_{rec} \mathcal{L}_{rec},$$  (9)

with $\lambda_{att} = \lambda_{rec} = 10, \lambda_{id} = 5$. The trainable modules of AEI-Net include the Multi-level Attributes Encoder and the AAD-Generator.

### 3.2. Heuristic Error Acknowledging Refinement Network

Although the face swap result $\hat{Y}_{s,t}$ generated with AEI-Net in the first stage can well retain target attributes like pose, expression and scene lighting, it often fails to preserve the occlusions appeared on the target face $X_t$. Previous methods [31, 30] address face occlusions with an additional face segmentation network. It is trained on face data containing occlusion-aware face masks, which require lots of manual annotations. Besides, such a supervised approach may hardly recognize unseen occlusion types.

![Figure 3: AEI-Net for the first stage. AEI-Net is composed of an Identity Encoder, a Multi-level Attributes Encoder, and an AAD-Generator. The AAD-Generator integrates informations of identity and attributes in multiple feature levels using cascaded AAD ResBlks, which is built on AAD layers.](image-url)
Figure 4: HEAR-Net for the second stage. \( \hat{Y}_{s,t} \) is the reconstruction of the target image \( X_t \), i.e., \( \hat{Y}_{s,t} = \text{AEI-Net}(X_t, X_t) \). \( Y_{s,t} \) is the swapped face from the first stage.

We proposed a heuristic method to handle facial occlusions. As shown in Figure 4(a), when the target face was occluded, some occlusions might disappear in the swapped face, e.g., the hair covering the face or the chains hang from the turban. Meanwhile, we observe that if we feed the same image as both the source and target images into a well-trained AEI-Net, these occlusions would also disappear in the reconstructed image. Thus, the error between the reconstructed image and its input can be leveraged to locate face occlusions. We call it the heuristic error of the input image, since it heuristically indicates where anomalies happen.

Inspired by the above observation, we make use of a novel HEAR-Net to generate a refined face image. We first get the heuristic error of the target image as

\[
\Delta Y_t = X_t - \text{AEI-Net}(X_t, X_t).
\]

Then we feed the heuristic error \( \Delta Y_t \) and the result of the first stage \( \hat{Y}_{s,t} \) into a U-Net structure, and obtain the refined image \( Y_{s,t} \):

\[
Y_{s,t} = \text{HEAR-Net}(\hat{Y}_{s,t}, \Delta Y_t).
\]

The pipeline of HEAR-Net is illustrated in Figure 4(b).

We train HEAR-Net in a fully self-supervised way, without using any manual annotations. Given any target face image \( X_t \), with or without occlusion regions, we utilize the following losses for training HEAR-Net. The first is an identity preservation loss to preserve the identity of the source. Similar as stage one, it is formulated as

\[
L'_{id} = 1 - \cos(z_{id}(Y_{s,t}), z_{id}(X_s)).
\]

The change loss \( L'_{chg} \) guarantees the consistency between the results of the first stage and the second stage:

\[
L'_{chg} = \| \hat{Y}_{s,t} - Y_{s,t} \|_2.
\]

The reconstruction loss \( L'_{rec} \) restricts that the second stage is able to reconstruct the input when the source and target images are the same:

\[
L'_{rec} = \begin{cases} \frac{1}{2} \| Y_{s,t} - X_t \|_2^2 & \text{if } X_t = X_s \\ 0 & \text{otherwise} \end{cases}.
\]

Since the number of occluded faces is very limited in most face datasets, we propose to augment data with synthetic occlusions. The occlusions are randomly sampled from a variety of datasets, including the EgoHands [3], GTEA Hand2K [15, 25, 24] and ShapeNet [9]. They are blended onto existing face images after random rotations, resizing and color matching. Note that we do not utilize any occlusion mask supervision during training, even from these synthetic occlusions.

Finally, HEAR-Net is trained with a sum of above losses:

\[
L_{\text{HEAR-Net}} = L'_{rec} + L'_{id} + L'_{chg}.
\]

4. Experiments

Implementation Detail: For each face image, we first align and crop the face using five point landmarks extracted with [11], the cropped image is of size 256 \( \times \) 256 covering the whole face, as well as some background regions. The number of attribute embeddings in AEI-Net is set to \( n = 8 \) (Equation 1). The number of downsamples/upsamples in HEAR-Net is set to 5. Please refer to the supplemental material for more details concerning the network structure and training strategies.

The AEI-Net is trained using CelebA-HQ [18], FFHQ [19] and VGGFace [34]. While the HEAR-Net is trained using only a portion of faces that have Top-10% heuristic errors in these datasets, and with additional augmentations of synthetic occlusions. Occlusion images are randomly sampled from the EgoHands [3], GTEA Hand2K [15, 25, 24] and object renderings from ShapeNet [9].

4.1 Comparison with Previous Methods

Qualitative Comparison: We compare our method with FaceSwap [2], Nirkin et al. [31], DeepFakes [1] and IPGAN [5] on the FaceForensics++ [36] test images in Figure 5. Comparison with the latest work FSGAN [30] is shown in Figure 6. We can see that, since FaceSwap, Nirkin et al., DeepFakes, and FSGAN all follow the strategy that first synthesizes the inner face region then blending it into the target face, as expected, they suffer from the blending inconsistency. All faces generated by these methods share exactly the same face contours with their target faces, and ignore the source face shapes (Figure 5 rows 1-4, Figure 6 rows 1-2). Besides, their results can not well respect critical information from the target image, such as the lighting (Figure 5 row 3, Figure 6 rows 3-5), the image resolutions (Figure 5 rows 2 and 4). IPGAN [5] suffers from decreased resolutions in all samples, due to its single-level attributes representation. IPGAN cannot well preserve expression of the target face, such as the closed eyes (Figure 5 row 2).
Our method addresses all these issues well. We achieve higher fidelity by well preserving the face shapes of the source (instead of the target), and faithfully respecting the lighting and image resolution of the target (instead of the source). Our method also has the ability to go beyond FS-GAN to handle occlusions.

Quantitative Comparison: The experiment is constructed on FaceForensics++ [36] dataset. For FaceSwap [2] and DeepFakes [1], the test set consists of 10K face images for each method by evenly sampled 10 frames from each video clip. For IPGAN [5], Nirkin et al. [31] and our method, 10K face images are generated with the same source and target image pairs as the other methods. Then we conduct quantitative comparison with respect to three metrics: ID retrieval, pose error and expression error.

We extract identity vector using a different face recognition model [41] and adopt the cosine similarity to measure the identity distance. For each swapped face from the test set, we search the nearest face in all FaceForensics++ original video frames and check whether it belongs to the correct source video. The averaged accuracy of all such retrievals is reported as the ID retrieval in Table 1, serving to measure identity preservation ability. Our method achieves higher ID retrieval score with a large margin.

We use a pose estimator [37] to estimate head pose and a 3D face model [10] to retrieve expression vectors. We report the $L$-2 distances of pose and expression vectors between the swapped face and its target face in Table 1 as the pose and the expression errors. Our method is advantageous in expression preservation while comparable with others in pose preservation. We do not use the face landmark comparison as [30], since face landmarks involve identity information which should be inconsistent between the swapped face and the target face.

Human Evaluation: Three user studies are conducted to evaluate the performance of the proposed model. We let the users select: i) the one having the most similar identity with the source face; ii) the one sharing the most similar head pose, face expression and scene lighting with the target image; iii) the most realistic one. In each study unit, two real face images, the source and the target, and four reshuffled face swapping results generated by FaceSwap [2], Nirkin et al. [31], DeepFakes [1] and ours, are presented. We ask users to select one face that best matches our description.

Table 1: Comparison on FaceForensics++ videos.

<table>
<thead>
<tr>
<th>method</th>
<th>ID retrieval</th>
<th>pose error</th>
<th>expression error</th>
</tr>
</thead>
<tbody>
<tr>
<td>DeepFakes [1]</td>
<td>81.96</td>
<td>4.14</td>
<td>2.57</td>
</tr>
<tr>
<td>FaceSwap [2]</td>
<td>54.19</td>
<td>2.51</td>
<td>2.14</td>
</tr>
<tr>
<td>Nirkin et al. [31]</td>
<td>76.57</td>
<td>3.29</td>
<td>2.33</td>
</tr>
<tr>
<td>IPGAN [5]</td>
<td>82.41</td>
<td>4.04</td>
<td>2.50</td>
</tr>
<tr>
<td>Ours</td>
<td>97.38</td>
<td>2.96</td>
<td>2.06</td>
</tr>
</tbody>
</table>

Table 2: User study results. We show the averaged selection percentages of each method.

<table>
<thead>
<tr>
<th>method</th>
<th>id</th>
<th>attr</th>
<th>realism</th>
</tr>
</thead>
<tbody>
<tr>
<td>DeepFakes [1]</td>
<td>13.7</td>
<td>6.8</td>
<td>6.1</td>
</tr>
<tr>
<td>FaceSwap [2]</td>
<td>12.1</td>
<td>23.7</td>
<td>6.8</td>
</tr>
<tr>
<td>Nirkin et al. [31]</td>
<td>21.3</td>
<td>7.4</td>
<td>4.2</td>
</tr>
<tr>
<td>Ours</td>
<td>52.9</td>
<td>62.1</td>
<td>82.9</td>
</tr>
</tbody>
</table>
4.2. Analysis of the Framework

Adaptive Embedding Integration: To verify the necessity of adaptive integration using attentional masks, we compare AEI-Net with two baseline models: i) Add: element-wise plus operations is adopted in AAD layers instead of using masks $M^k$ as in Equation 5. The output activation $h_{out}^k$ of this model is directly calculated with $h_{out}^k = A^k + I^k$; ii) Cat: element-wise concatenation is adopted without using masks $M^k$. The output activation becomes $h_{out}^k = \text{Concat}[A^k, I^k]$. Results of the two baseline models, as well as the AEI-Net, are compared in Figure 7. Without a soft mask for fusing embeddings adaptively, the faces generated by baseline models are relatively blurry and contain lots of ghosting artifacts.

We also visualize the masks $M^k$ of AAD layers on different levels in Figure 8, where a brighter pixel indicates a higher weight for identity embedding in Equation 5. It shows that the identity embedding takes more effect in low level layers. Its effective region becomes sparser in middle levels, where it activates only in some key regions that strongly relates to the face identity, such as the locations of eyes, mouth and face contours.

Multi-level Attributes: To verify whether it is necessary to extract multi-level attributes, we compare with another baseline model called Compressed, which shares the same network structure with AEI-Net, but only utilizes the first three level embeddings $z_{att}^k, k = 1, 2, 3$. Its last embedding $z_{att}^3$ is fed into all higher level AAD integrations. Its results are also compared in Figure 7. Similar to IPGAN [5], its results suffer from artifacts like blurriness, since a lot of attributes information from the target images are lost. To understand what is encoded in the attributes embedding, we concatenate the embeddings $z_{att}^k$ (bilinearly up-sampled to $256 \times 256$ and vectorized) from all levels as a unified attribute representation. We conduct PCA to reduce vector dimensions as 512. We then perform tests querying faces from the training set with the nearest $L_2$-2 distances of such vectors. The three results illustrated in Figure 9 verify our intention, that the attributes embeddings can well reflect face attributes, such as the head pose, hair color, expression and even the existence of sunglasses on the face. Thus it also explains why our AEI-Net sometimes can preserve occlusions like sunglasses on the target face even without a second stage (Figure 10(8)).

Second Stage Refinement: Multiple samples are displayed with both one-stage results $\hat{Y}_{s,t}$ and two-stage results $Y_{s,t}$ in Figure 10. It shows that the AEI-Net is able to generate high-fidelity face swapping results, but sometimes its output $\hat{Y}_{s,t}$ does not preserve occlusions in the target. Fortunately, the HEAR-Net in the second stage is able to recover them. The HEAR-Net can handle occlusions of various kinds, such as the head (1), hand (2), hair (3), face painting (4), mask (5), translucent object (6), eyeglasses (7), headscarf (8) and floating text (9). Besides, it is also able to correct the color-shift that might occasionally happen in $\hat{Y}_{s,t}$ (10). Moreover, the HEAR-Net can help rectify the face shape when the target face has a very large pose (6).

4.3. More Results on Wild Faces

Furthermore, we demonstrate the strong capability of FaceShifter by testing wild face images downloaded from the Internet. As shown in Figure 11, our method can handle face images under various conditions, including large

<table>
<thead>
<tr>
<th>Source</th>
<th>Target</th>
<th>Add</th>
<th>Cat</th>
<th>Compressed</th>
<th>AEI-Net</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image1.png" alt="Source Image" /></td>
<td><img src="image2.png" alt="Target Image" /></td>
<td><img src="image3.png" alt="Add Image" /></td>
<td><img src="image4.png" alt="Cat Image" /></td>
<td><img src="image5.png" alt="Compressed Image" /></td>
<td><img src="image6.png" alt="AEI-Net Image" /></td>
</tr>
</tbody>
</table>

Figure 7: Comparing AEI-Net with three baseline models. The two models Add and Cat are for ablation studies of the adaptive embedding integration. The model Compressed is for ablating multi-level attributes representation.

For each user, 20 face pairs are randomly drawn from the 1K FaceForensics++ test set without duplication. Finally, we collect answers from 100 human evaluators. The averaged selection percentage for each method on each study is presented in Table 2. It shows that our model surpasses the other three methods all in large margins.
4.4. Examining Forged Face Detection Algorithms

Finally, we examine the performance of different face forgery detection algorithms on our face swapping results. First, we randomly generate 5,000 face swapping images and 5,000 real images. Then we apply the model from FF++ [36] and Face X-Ray [23] and show the detection results in Table 3. We can notice that Face X-Ray has impressive results on our generated images.

5. Conclusions

In this paper, we proposed a novel framework named FaceShifter for high fidelity and occlusion aware face swapping. The AEI-Net in the first stage adaptively integrates the identity and attributes for synthesizing high fidelity results. The HEAR-Net in the second stage recovers anomaly region in a self-supervised way without any manual annotations. The proposed framework shows superior performance in generating realistic face images given any face pairs without subject specific training. Extensive experiments demonstrate that the proposed framework significantly outperforms previous face swapping methods, setting up a new benchmark for face forensics researches.

<table>
<thead>
<tr>
<th>Methods</th>
<th>AUC</th>
<th>AP</th>
<th>EER</th>
</tr>
</thead>
<tbody>
<tr>
<td>FF++ [36]</td>
<td>52.22</td>
<td>52.87</td>
<td>0.4805</td>
</tr>
<tr>
<td>Face X-Ray [23]</td>
<td>96.82</td>
<td>90.53</td>
<td>0.0956</td>
</tr>
</tbody>
</table>

Table 3: Results in terms of AUC, AP and EER for FF++ [36] and Face X-ray [23] on our generated faces.
References


