FACETRACER: Unveiling Source Identities from Swapped Face Images and Videos for Fraud Prevention

Zhongyi Zhang, Jie Zhang, Wenbo Zhou, Xinghui Zhou, Qing Guo, Weiming Zhang, Tianwei Zhang, and Nenghai Yu

Abstract—Face-swapping techniques have advanced rapidly with the evolution of deep learning, leading to widespread use and growing concerns about potential misuse, especially in cases of fraud. While many efforts have focused on detecting swapped face images or videos, these methods are insufficient for tracing the malicious users behind fraudulent activities. Intrusive watermark-based approaches also fail to trace unmarked identities, limiting their practical utility. To address these challenges, we introduce FACETRACER, the first non-intrusive framework specifically designed to trace the identity of the source person from swapped face images or videos. Specifically, FACETRACER leverages a disentanglement module that effectively suppresses identity information related to the target person while isolating the identity features of the source person. This allows us to extract robust identity information that can directly link the swapped face back to the original individual, aiding in uncovering the actors behind fraudulent activities. Extensive experiments demonstrate FACETRACER's effectiveness across various face-swapping techniques, successfully identifying the source person in swapped content and enabling the tracing of malicious actors involved in fraudulent activities. Additionally, FACETRACER shows strong transferability to unseen face-swapping methods including commercial applications and robustness against transmission distortions and adaptive attacks.

Index Terms—DeepFake, Fraud Prevention, Identity Tracing

1 INTRODUCTION

FACE swapping is a well-known technique for deepfake generation, which can seamlessly transfer the identity from the target image to the source image, while the facial attributes of the source image hold intact. In recent years, popular commercial platforms like Snapchat [1] and FaceApp [2] have made face swapping accessible to millions of users worldwide, while open-source projects like DeepFaceLab [3] and FaceSwap [4] have fostered a thriving community of developers and enthusiasts. With the immense popularity and interest in face swapping, this technique has been widely used in various industries such as movie production and entertainment applications. Recently, there emerge new face-swapping solutions that can even support seamless face-swapping in real time [5].

However, along with the commercial value and practical applications of face-swapping technology comes a significant risk. Recently, face-swapping techniques have been widely exploited in financial scams. In such cases, verifying the other party's identity in person is often difficult, leading many to rely on video calls for confirmation. Fraudsters, however, have begun using face-swapping technology to impersonate victims' family members or superiors during these calls, exploiting the trust they build to carry out scams, such as pressuring victims to transfer money [6] or luring them into fake investment schemes [7]. In January 2024, according to Hong Kong police, a finance worker at a multinational firm was tricked into paying out $25 million to fraudsters using face-swapping technology to pose as the company's chief financial officer in a video conference call [8]. These incidents highlight the growing concern surrounding the potential misuse of face-swapping technology and the need for countermeasures to prevent such malicious activities.

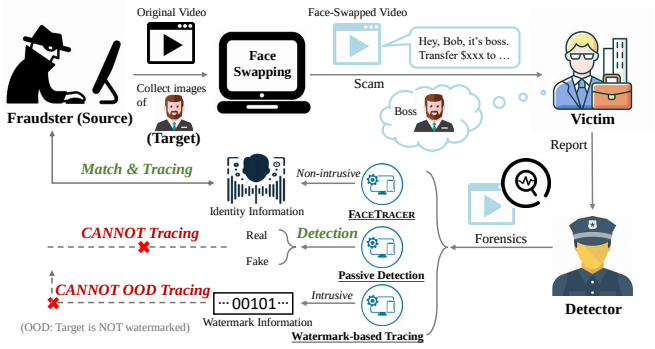


Fig. 1. FACETRACER can achieve non-intrusive tracing to unveil the fraudster (source) identities for effective forensics.

As illustrated in Figure 1, a fraudster adopts face-swapping to replace their (source) identity with the iden-

- *Zhongyi Zhang, Wenbo Zhou, Xinghui Zhou, Weiming Zhang, and Nenghai Yu are with the School of Cyber Science and Technology, University of Science and Technology of China, Hefei, Anhui 230026, China (E-mail: ericzhang@mail.ustc.edu.cn; welbeckz@ustc.edu.cn; zhouxinghui@mail.ustc.edu.cn; zhangwm@ustc.edu.cn; ynh@ustc.edu.cn).*
- *Jie Zhang and Qing Guo are with Centre for Frontier AI Research, Agency for Science, Technology and Research (A*STAR), Singapore (E-mail: zhang_jie@cfar.a-star.edu.sg; guo_qing@cfar.a-star.edu.sg).*
- *Tianwei Zhang is with College of Computing and Data Science at Nanyang Technological University (E-mail: tianwei.zhang@ntu.edu.sg).*
- *Wenbo Zhou is the corresponding author.*

tity of the target person (*e.g.*, the victim's boss) to commit fraud. To prevent such misuse and hold the fraudster legally accountable, it is crucial to uncover their true identity. However, most existing works [9], [10], [11], [12] focus only on detecting whether images or videos have been face-swapped, without addressing the challenge of tracing the malicious user. Very recently, some watermark-based methods [13], [14] have been proposed to trace the creators of swapped face images or videos, but they need to intrusively embed watermark information into the source images or videos before the incident occurs. As a result, these approaches cannot generalize to out-of-distribution (OOD) identities (*i.e.*, unwatermarked identities), limiting their effectiveness in real-world applications.

While malicious attackers may download face images from the Internet as the source identity to bypass the identity tracing, in fraud scenarios on we focus in this paper, there are two significant drawbacks to this approach: i) These images or videos can be found and used as references to verify if the content is fake. ii) These materials may contain watermarks, which can also be used to identify manipulated content. To maintain greater flexibility and control over the swapped face images or videos, attackers often need to capture new images or videos themselves to serve as the *source* for face swapping. Therefore, if we unveil the source identity from the manipulated content, we can trace the malicious fraudster.

For this, we introduces FACETRACER, the first non-intrusive tracing framework designed to effectively extract the identity information of the source person from swapped face images or videos. As shown in Figure 1, once the offense of abusing face-swapping methods occurs, FACETRACER can assist law enforcement officers in tracing the source person. To achieve it, there are two primary challenges: 1) In practice, attackers may replace their original identity with various faces, resulting in swapped face images or videos visually resembling multiple individuals. In addition, the identities of the attackers can be diversified, treating this task as a classification problem by simply relabeling single swapped face images or videos is impractical. To address it, we opt to ❶ *extract the identity information* with a neural network, which is trained on a large-scale face-swapping dataset comprising over 1 million images across 30,000 identities [15]. 2) As demonstrated in many previous studies [10], [16], the identity information conveyed by swapped face images or videos is a hybrid of the source and target person's identities, dominated by the target person's identity. Thus, it is crucial to eliminate the influence of the target-related identity information and capture the source-related identity information. To overcome this, we design an ❷ *identity information disentanglement module*, targeting at retaining only the source-related identity information while removing target-related identity information.

Empowered by the above two designs, FACETRACER can trace the attacker by comparing the similarity between the extracted source identity information and the identity information stored in **the identity pool**. Notably, in the real-life forensics process we face an open-world problem, where the information relevant to the attacker may not be presented in the training data. Even under such cases, FACETRACER is still able to extract the identity information of the attacker.

Extensive experiments are conducted over four popular face-swapping methods that explicitly separate identity and attribution information of faces: HiRes [17], FaceShifter [18], SimSwap [19], and InfoSwap [20]. FACETRACER demonstrates its superior ability to extract identity information of the source person from swapped face images or videos under different forensic conditions, and has good transferability for identities and methods that have not appeared in the training process. In addition, evaluation on other two face-swapping methods (*i.e.*, MegaFS [21] and DiffSwap [22]) shows that FACETRACER also achieves good performance against face-swapping solutions that do not explicitly separate attribution and identity information. Notably, FACETRACER performs well on commercial apps such as Faceover [23] and DeepFaker [24]. We also investigated the robustness of FACETRACER against distortions that may be encountered during various network transmissions, such as JPEG, color jittering, etc. Saliency map visualization is also conducted to better understand the regions FACETRACER focuses on during tracing. Finally, we further analyze the impact of different backbones, the disentangle networks, and the scale of the identity pool, adaptive evaluation on FACETRACER is also discussed.

In a nutshell, our main contributions could be summarized as follows:

- We proposed the first non-intrusive tracing framework, FACETRACER, which can extract the source identity information from the swapped face images or videos, enabling effective forensics.
- FACETRACER consists of two main parts: an identity extractor trained on large-scale dataset and an effective disentangle network to maximally eliminate the influence from the target-related identity information. Based on these, FACETRACER can extract identity information that is highly related to the source person.
- FACETRACER holds general effectiveness among different face-swapping methods, including four that explicitly and two that implicitly disentangle identity and attribute information, and two commercial apps, even those that have not been seen during the training phase.
- FACETRACER is robust against different distortions and even adaptive attacks, making it effective in practice.

2 BACKGROUND

2.1 Face Swapping

In the study of the human face, the entirety of facial information can be categorized into two main components: identity information, determining "who this person is", and facial attributes, encompassing expressions, hairstyles, gaze direction, etc. The objective of face swapping is to make the person in an image or a video resemble the target person while retaining the facial attributes of the source person unchanged. This can lead others to mistakenly believe that the person in the image or video is the target person. Figure 2 describes the process of traditional face-swapping methods in general. Formally, an identity encoder $\mathcal{E}_{id}$ aims to extract the identity embedding $x_{src,id}$ and $x_{tar,id}$ from the source image and the target image, respectively. Similarly, we can obtain the attribution embedding $x_{src,attr}$

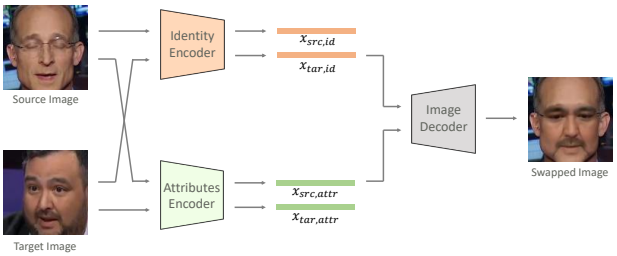


Fig. 2. Illustration of traditional face swapping techniques.

and $x_{tar,attr}$. Then, feeding $x_{src,attr}$ and $x_{tar,id}$ into the face image decoder $\mathcal{D}$ together, the swapped image is acquired.

It is noted that achieving the disentanglement of identity and attribute information plays a crucial role during the face swapping process. There are some methods that aim to explicitly disentangle the identity and attribute information with well-designed networks. For example, some methods [25], [26] leverage off-the-shelf 3D-based models [27], [28], [29], [30] to extract facial attributes, which only remain the structure of human faces and neglect the corresponding texture information. Similarly, HiRes [17] performs face-swapping by leveraging the generative networks and 2D attributes extraction methods. These methods separately extract identity and attribute information with off-the-shelf networks and combine them to produce swapped face results. Besides, some other methods [18], [19], [20], [31], [32] first train a single conditional GAN to reconstruct face images with two branches before producing face-swapping, namely, the identity extraction branch and the attributes extraction branch. That is to say, the disentanglement of identity and attribute information is accomplished through the two branches.

Recently, as the capability of generative networks increases, some methods [21], [22], [33] have also adopted approaches that do not explicitly disentangle identity and attributes. For example, MegaFS [21] assumes that some existing generative networks can inherently separate identity and attribute information, *i.e.*, different latent space layers of StyleGAN [34], [35] correspond to different levels of semantics from coarse to fine. Moreover, based on the success of diffusion models [36], [37], DiffSwap [22] leverages the latent space of diffusion models as it preserves the layout of the source image. Instead of training the network to individually extract the facial attributes, these methods directly inject the target identities $x_{tar,id}$ into the generative network to generate the swapped faces. In Section 5.1, we adopt 4 open-sourced face-swapping methods that explicitly disentangle identity and attribute information (*i.e.*, HiRes [17], FaceShifter [18], SimSwap [19], and InfoSwap [20]) and two open-sourced face-swapping methods that implicitly disentangle such information (*i.e.*, MegaFS [21] and DiffSwap [22]) for comprehensive evaluation.

2.2 Identity Extraction

In our scenario, we need to extract the identity information from the suspected face image at the first step. In the process of extracting face identity from an input face image, the initial step involves detecting faces within the image using face detection techniques like MTCNN [38] or Face Attention Network (FAN) [39]. Once detected, bounding boxes are used to isolate these faces. Subsequently, the faces are cropped and resized to a standardized size, typically 112×112 pixels. Afterwards, the cropped and scaled faces I are fed to the identity extraction network. The majority of current identity extraction networks employ the ResNet [40] family as their backbone architecture, while some networks also utilize the Vision Transformer (ViT) [41] as an alternative backbone. The same thing is that the identity information is usually represented as the output of the network, typically a 512-dimensional real vector. Thus, an identity information extraction network could be formulated as:

$$\mathcal{F} : I \rightarrow x \in \mathbb{R}^{1\times 512}. \tag{1}$$

While various identity information extraction networks may share similar or identical backbones, the final representation of x can be different significantly due to the diverse training strategies. Briefly, training an open-set identity extraction network can be regarded as a classification task. Intuitively, the simplest identity extraction network consists of a softmax activation with a classification layer, employing the cross-entropy loss function to constrain the representation distribution of extracted identity information, which could be formulated as:

$$\mathcal{L} = -\frac{1}{N}\sum_{i=1}^{N} y_i \log \frac{e^{\boldsymbol{W}_{\boldsymbol{y_i}}^T x_i}}{\sum_{j=1}^{N} e^{\boldsymbol{W}_{\boldsymbol{j}}^T x_i}}, \tag{2}$$

where x_i denotes the real identity vector extracted from the network, and $\boldsymbol{W_i}$ denotes the linear mapping layer that converts the identity vector to a prediction of the likelihood that the current identity vector belongs to the label i. However, advancements in this field have introduced several techniques to enhance performance by modifying the loss function and decision boundaries. For instance, SphereFace [42] introduces a multiplicative angular margin, CosFace [43] introduces an additive cosine margin, ArcFace [44] introduces an additive angular margin, and AdaFace [45] introduces adaptive angular margin. These methods aim to make the extracted identity features more compact in the angular feature space, ultimately improving the performance of identity extraction.

2.3 Feasibility of Extracting the Source Identity

FACETRACER aims to extract the source identities $x_{src,id}$ from the swapped face images or videos, and the feasibility of this goal can be attributed to two main reasons.

First, current methods are unable to perfectly disentangle identity and attribute information, which means that the extracted source attributes $x_{src,attr}$ contains some source identity information. Therefore, when performing face-swapping operation, part of the source identity information will be inevitably retained. Below, we adopt the state-of-the-art attributes extraction network EMOCA [30] and conduct experiments to verify that the extracted attribute information inherently contains some identity information. Here, we consider a special case, wherein all faces share the very similar attributions but different identity information. In consideration of the difficulty of collecting data with the

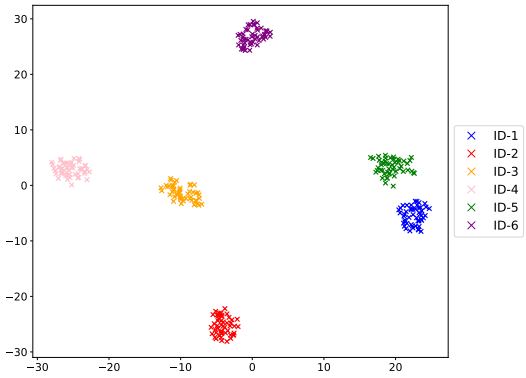


Fig. 3. T-SNE visualization of the attribution embedding extracted by EMOCA [30] from the generated images (the same attributions but different identities). The distinguishable cluster verifies that some identity information is preserved in the extracted attribution embedding.

same attributes in reality, we utilize StyleGAN [34], [35] to demonstrate this. It is known that different latent layers in StyleGAN control different levels of the semantic information [46], [47]. Thus, we fixed the first 12 layers of the latent codes in a standard 18-layer StyleGAN network to produce images with similar attributes but different identities. For more images, we fix the latent codes and adjust the noise injected to StyleGAN, and visual examples of the generated images are provided in the supplementary material. Finally, we use EMOCA [30] to extract the attributes from the above generated images, and find that the expression cosine similarity between different identities is greater than 0.9. Thereafter, we performed t-SNE analysis on the expression embedding extracted from 6 different identities, within each identity 50 images were generated by adjusting the noise input. The analysis result of expression embedding is presented in Figure 3, where a clear clustering exists between the different identities with highly similar attribute. In addition, we trained an SVM to classify these data and obtained over 99% classification accuracy. All the results consistently verify that the extracted attribute information contains some identity information.

Second, another important step in the face-swapping is the extraction of the target identity information $x_{tar,id}$. However, this process is also imperfect. To illustrate this, we randomly selected several videos with single faces (*i.e.*, the same identity) and utilized the widely-used identity extraction network, ArcFace [44], to extract the identity information. We then calculated the cosine identity similarity between each frame with different poses and expressions, obtaining a cosine identity similarity of approximately 0.8. This observation highlights the imperfect nature of identity information extraction. Due to this imperfection, the identity in the swapped face result cannot be seamlessly replaced with the identity of the target person. In general, the identity in the swapped face result will behave as a hybrid identity that is a mixture of most of the target identity and a small portion of the source identity.

2.4 Face Identification

Face identification systems are vital tools for the tracing task, whose objective is to identify the most similar identity from an input face image I to one of the identities within an existing identity pool holding N identities, *i.e.*, $\{x_n\}_{n\in N}$. This process enables the determination of the specific person in the pool corresponding to the input face image. Typically, we use the cosine similarity metric $cos(\cdot,\cdot)$ to measure the similarity of different identity information, so face identification system can be formalized as follows:

$$\arg\max_{n} cos(\mathcal{F}(I), x_n). \tag{3}$$

However, face-swapping techniques can mislead face identification systems, making it believe that the identity in the image belongs to someone else, leading to malicious events such as fraudulent and scapegoating. In this paper, FACETRACER first extract the source identity information from the suspected face, and then compare the it with the one in the identify pool for tracing the malicious user.

3 THREAT MODEL

In this section, we first give the problem formulation, and then clarify the ability and goals of both *Attacker* and *Defender*. Without loss of generality, we refer to the party who uses face-swapping methods to convert their identity to the target identity as the *Attacker* and the party who wants to extract the source identity as the *Defender*. Besides, three practical scenarios are also illustrated for subsequent evaluation.

3.1 Problem Formulation

Most contemporary face-swapping methods primarily operate at the image level, although some extend to video processing, these methods typically swap faces in each frames individually before assembling them into a cohesive video sequence. Consequently, our discussion predominantly centers on swapped face images. Nevertheless, in Section 5.6, we delve into the impact of varying input frame counts when extending FACETRACER to video level.

Given a raw facial image I_{src} of the source person, the swapped face image I_{swap} was generated through face-swapping method to mimic the identity of the target person $x_{tar,id}$, *i.e.*,

$$I_{swap} = \mathcal{S}(I_{src}, I_{tar}), \tag{4}$$

where $\mathcal{S}(\cdot,\cdot)$ denotes arbitrary face-swapping methods. As shown in Figure 4, face-swapping methods aim to significantly increase the identity similarity between I_{swap} and I_{tar}, *i.e.*,

$$cos(\mathcal{F}(I_{swap}), \mathcal{F}(I_{tar})) \to 1, \tag{5}$$

while pursuit the decrease of the identity similarity between I_{swap} and I_{src}, *i.e.*,

$$cos(\mathcal{F}(I_{swap}), \mathcal{F}(I_{src})) \to 0, \tag{6}$$

making it difficult for the defender to unveil the identity of the attacker. As we have discussed in Section 2.3, due to the imperfectness of the face-swapping process, the swapped face result will form a hybrid identity that is a mixture of both the source and target identities.

In this paper, FACETRACER is intended to extract the source identity information from the suspect face image I, *i.e.*,

$$\mathcal{E} : I \to x_{src,id} \in \mathbb{R}^{1\times 512}. \tag{7}$$

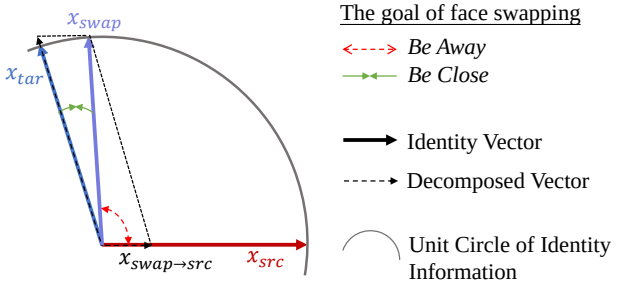


Fig. 4. Relationships between identity information during face-swapping. All identity information is normalized and located on the unit circle. The identity information of the swapped face image is dominated by the target identity information and still contains some source identity information.

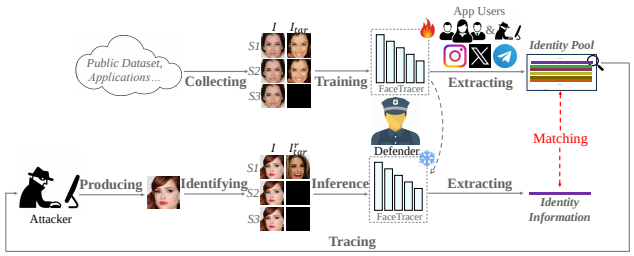


Fig. 5. Illustration of three scenarios of FACETRACER. Different Scenarios differ in the access to the reference images I_r during the training and testing phases. The reference images I_r denote face images of the target person used as reference. The blank images are used in the absence of reference images.

When the fed image is an un-swapped face, the goal of Eq. (7) is the same with Eq. (1). If the suspected face image is a swapped one, $\mathcal{E}$ focuses on extracting identity of the source person (*i.e.*, the malicious user), while $\mathcal{F}$ targets extracting the general identity of the swapped image I_{swap} itself. In the following part, we adopt the straightforward solution $\mathcal{F}$ as the baseline for comparison.

3.2 Attacker's Ability and Goal

We assume that the attacker possesses facial images $\{I^a_{tar}\}$ of the target person, and can extract target person's identity information $x_{tar,id}$ from them. The attacker then manipulates carefully crafted images I_{src} or videos V_{src}, employing face-swapping methods to replace their faces within these media with the face of the target person. This process aims to create swapped face results I_{swap} or V_{swap}. It is worth noting that, due to the design that explicit disentangle identity and attribute information used in most of the current face-swapping methods, we assume that the model used by the attacker adopts this design. Besides, we also evaluate our transferability to face-swapping methods that do not explicitly disentangle identity and attribute information, and even commercial apps in Section 5.3.

3.3 Defender's Ability and Goal

We assume the defender acquires an image I or video V, recognizing it as potentially face-swapped but unaware of the face-swapping method used in I or V. The defender may also obtain some reference face images of the target person $\{I^r_{tar}\}$, which allows the extraction of the target person's identity information $\hat{x}_{tar,id}$. Notably, these acquired face images do not need to be those held by the attacker (*i.e.*, $\{I^r_{tar}\} \neq \{I^a_{tar}\}$) and $\hat{x}_{tar,id}$ need not be exactly the same as $x_{tar,id}$.

The defender's objective is to train a FACETRACER model that could extract the identity information of the source person from the input image. After completing the training phase of FACETRACER, the defender can obtain an identity pool for identity matching, which is practical in reality, *e.g.*, the Police Force could use all the face images in their database to create such an identity pool. To preserve privacy, only identity embeddings rather than face images are stored in the identity pool. Subsequently, the defenders could obtain the identification network to match extracted identity information with those in the identity pool, which offers the possibility to trace the attacker. We posit that FACETRACER, at its core, functions as a specialized form of face recognition. Therefore, privacy-preserving methods such as adding adversarial noise [48] can be seamlessly integrated into our training process. This integration would enhance FACETRACER's privacy protection capabilities without compromising its effectiveness in detecting face-swapping fraud.

For forensics integrity, the defender shall extract the correct identity information from un-swapped faces, which is identical to the true identity information of these images or videos.

3.4 Practical Scenarios

In practice, the defender may have access to some reference images of the target person to support forensics (*i.e.*, $\{I^r_{tar}\}$), and Eq. (7) can be specified as follows:

$$\max cos(\mathcal{E}(I, *|\theta), x_{src,id}), \tag{8}$$

where θ is the parameters of the FACETRACER model and $*$ denotes the options of the reference images. Based on the defender's access to reference images during the training and inference phases, we classify FACETRACER into three main scenarios: full-reference scenario, half-reference scenario and none-reference scenario, as shown in Figure 5.

Note: In all three scenarios, there is no identity overlap between the training and inference phase. That is to say, FACETRACER learns to disentangle source and target identity rather than memorizing some identities.

Full-reference Scenario (S1): In full-reference scenario, the target person's face images $\{I_{tar}\}$ used for creating swapped face images or videos are available during the training phase, and reference images of the target person $\{I^r_{tar}\}$ are also available during the inference phase. Furthermore, to ensure that FACETRACER can correctly extract identity information from the images without face-swapping, some raw images without face-swapping are used during the training phase, where the corresponding reference images are replaced by the blank images of the same size. In this scenario, defenders like law enforcement

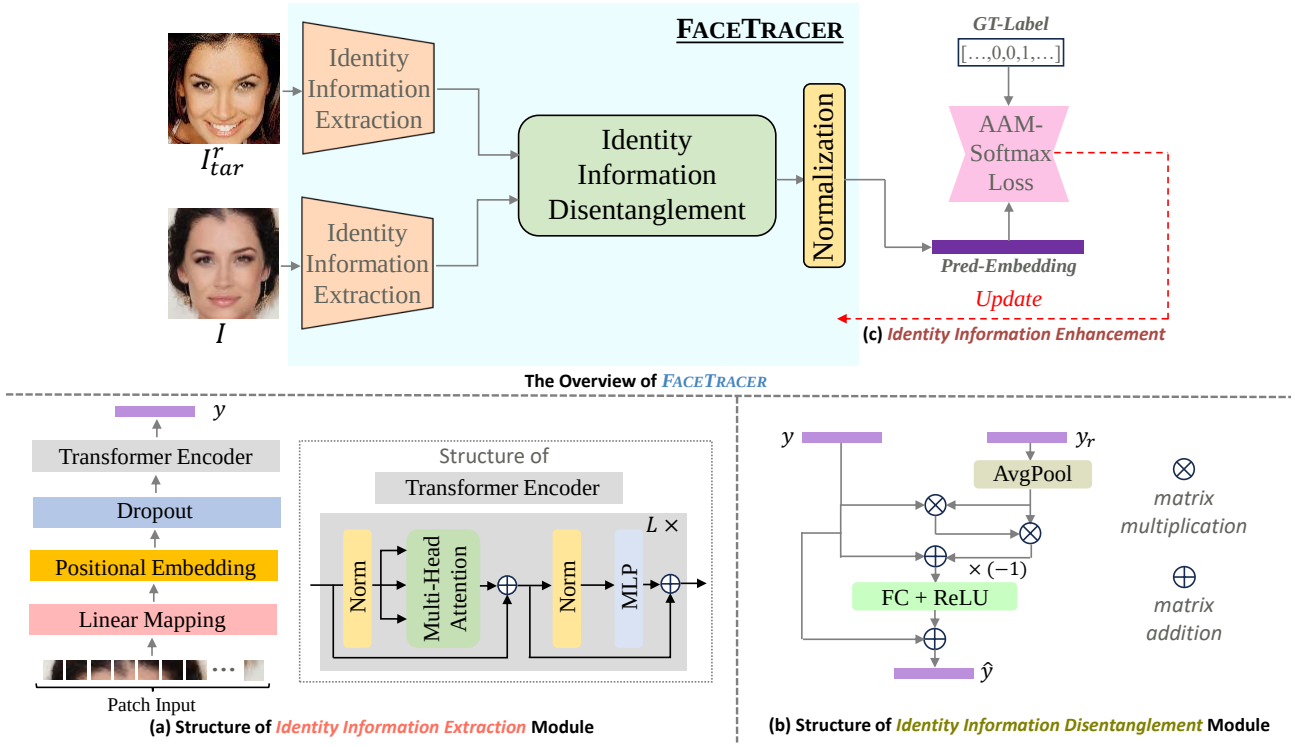


Fig. 6. FACETRACER consists of three designs: a) *identity information extraction* module aims to extracts crude identity embedding from the input image, b) *identity information disentanglement* module eliminates the influence of the identity information from the target person, and c) *identity information enhancement* module further enhances the discriminative capability. The red dotted line indicates the optimization of FACETRACER with the identity information enhancement module, which is also known as the AAMSoftmax loss in Eq. (12).

agencies have identified the specific target identities used in the face-swapping process. This situation typically arises during retrospective investigations.

Case Example: Consider a case where an attacker creates a fraudulent video by swapping the face of the victim's boss. Once the victim reports the fraud to authorities, the defender can easily determine the target person and gather reference images. This knowledge of the target identity significantly aids in the investigation and analysis of the face-swapped content.

Half-reference Scenario (S2): The main difference between S2 and S1 is that the reference images of the target person $\{I^r_{tar}\}$ are unavailable during the inference phase. The training phase of S2 is identical to that of S1. During inference phase, the reference images $\{I^r_{tar}\}$ are replaced with blank images of the same size. In this scenario, the defender is not determined who exactly the victim is and therefore uses a blank image as a substitute for the target face image in the inference phase. As FACETRACER learns to extract source-relevant identity information in the swapped face images, the reference images in the inference phase are helpful but not essential.

None-reference Scenario (S3): In none-reference scenario, not only $\{I^r_{tar}\}$ are unavailable during the inference phase, but $\{I_{tar}\}$ are also unavailable during the training phase, wherein reference images are replaced with blank images of the same size in both the training and inference phases. While FACETRACER obtain less information during the training phase, Table 2 shows that reference images are not necessary to the training phase. In other words, FACETRACER can disentangle the source and target identity well even without the reference information of the target identity.

4 FACETRACER

The purpose of FACETRACER is to design an identity information extractor of the source person for any input image. To achieve this, we referred to previous work [49] on model designs for extracting identity information and further designed a disentangle module to extract the identity information of the source person rather than hybrid identity information. We also utilized a normalization layer followed by an additive angular margin softmax activation to enhance the performance of FACETRACER. Noted that the normalized output will be treated as the extracted identity information of the source person. The training phase of FACETRACER consists of three designs:

- ***Identity Information Extraction*** module $\mathcal{E}_1(\cdot)$ extracts the crude identity information from the input image.
- ***Identity Information Disentanglement*** module $\mathcal{E}_2(\cdot)$ is fed with the extracted identity information. Then, it cleanse the relevant portion of the target person's identity information from the hybrid identity information and retain the identity information of the source person.
- ***Identity Information Enhancement*** is a customized loss $\mathcal{L}$ that further reduces the distance between the extracted identity information and the ground-truth identity information, and increases the distance between the generated identity information and other identity information, thus enhancing the discriminative ability of facial identification systems.

In a nutshell, FACETRACER can be formulated as $\mathcal{E}(I, *|\theta) = \mathcal{E}_2(\mathcal{E}_1(I), *)$, trained with loss function $\mathcal{L}$. Figure 6 show-

cases the designs of FACETRACER, and we will describe them in details below.

4.1 Identity Information Extraction

We leverage the ViT-S model from the vision transformer (ViT) family [41] as the default backbone for identity information extraction. As shown in Figure 6 (a), due to the patch-based input structure of the Transformer, the input image is first divided into $s \times s$ patches. After obtaining the patches, they are mapped to d-dimensional token vectors through a linear mapping layer, resulting in a total of n d-dimensional tokens, where n is calculated from $n = \lfloor \frac{h*w}{s*s} \rfloor$. Subsequently, positional embedding is applied to these patches followed by a dropout layer with the dropout rate p to enhance the model's performance. ViT-S comprises l transformer blocks in total, each consisting of two parts: (i) a multi-head self-attention layer with normalized input and the skip-connection mechanism and (ii) a 2-layer MLP with normalized input and the skip-connection mechanism. The input and output of each Transformer block are the same, and the self-attention mechanism offers the capability to extract identity information from the image. Generally, the identity information extraction block converts the input image into an $n \times d$-dimensional identity information vector. We formulate the identity information extraction module as:

$$\boldsymbol{y} = \mathcal{E}_1(I) \quad \boldsymbol{y} \in \mathbb{R}^{n*d}, \forall I. \tag{9}$$

The hyper-parameters s, d, h, w, p, and l are set to 9, 512, 112, 112, 0.1, and 12 by default in our framework.

As discussed in Section 2.2, utilizing ResNet [40] for identity information extraction remains a viable approach. While the ViT family has demonstrated superior performance in various tasks, many identity information extraction networks still opt for the ResNet as the backbone, exemplified by ArcFace [44]. Unlike the ViT series, ResNet directly processes the image as input to the network, extracting high-level semantic information through several residual blocks. In Section 5.7, we will explore the implications of replacing the backbone with ResNet18 on the model's performance.

4.2 Identity Information Disentanglement

The identity information disentanglement module is the key to extract the identity information of the source person from the input image. As mentioned in Section 3.1 and Figure 4, the swapped face image has identity information that is similar to the identity information of the target person but is distant from that of the source person. Therefore, simply utilizing an identity extraction network $\mathcal{E}_1$ to extract the identity information of input images does not effectively obtain the identity information of the source person. To address this issue, we aim to explicitly eliminate the influence of the target person's identity information by designing a dual-input network. This approach maximizes the relevance between the extracted identity information and the identity information of the source person. Inspired by REVELIO [50], as shown in Figure 6 (b), we designed a disentanglement module to explicitly disentangle identity information. Under this design, the inputs of this module are $\boldsymbol{y}$ and $\boldsymbol{y}_r$, calculated from the following equations:

$$\boldsymbol{y} = \mathcal{E}_1(I), \quad \boldsymbol{y}_r = \mathcal{E}_1(I^r_{tar}), \tag{10}$$

where I denotes the suspected input image and I^r_{tar} denotes the reference image of the target person, noted that I^r_{tar} could be blank image in scenarios S2 and S3. Because the original identity information vector $\boldsymbol{y}$ is added in a residual learning fashion, the total identity information disentanglement module $\mathcal{E}_2$ could be formulated as:

$$\hat{\boldsymbol{y}} = \mathcal{E}_2(\mathcal{E}_1(I), *). \tag{11}$$

In addition, in distinction to explicitly disentangle the identity information of the target person from the extracted hybrid identity information, it is also viable to use some implicit disentanglement methods such as cross-attention. We will discuss the impact of this implicit disentangle approach on the model performance in Section 5.7.

4.3 Identity Information Enhancement

As mentioned in Section 3.1, conventional identity extraction networks typically employ a normalized softmax activation with a cross-entropy loss function. However, relying solely on this setting may not adequately differentiate between extracted identity information, potentially impacting the identification model's performance. To address this, we took inspiration from ArcFace [44] and implemented an additive angular margin softmax (AAM-Softmax) activation. This approach aims to better separate the extracted identity information of different source person on the unit circle. The loss function of FACETRACER is formulated as:

$$\mathcal{L} = -\frac{1}{N}\sum_{i=1}^{N} y_i \log \frac{e^{s(cos\phi_{y_i}+m)}}{e^{s(cos\phi_{y_i}+m)} + \sum_{j=1,j\neq y_i}^{N} e^{scos\phi_j}}, \tag{12}$$

where $x_i = \mathcal{E}(I_i, *|\theta)$ is the 512-dimensional real identity vector output of the previous network and $cos\phi_j$ is calculated from:

$$cos\phi_j = \frac{\boldsymbol{W}_j^T x_i}{||\boldsymbol{W}_j|| * ||x_i||} \quad \forall j, \forall x_i. \tag{13}$$

Here s and m are parameters that control the distribution of extracted identity information vectors, we set s and m to 64 and 0.5 by default. Additionally, while the backbones of the extraction model may vary in structure, they all produce outputs of the same size: 512-dimensional real vectors. Therefore, we apply the AAM-Softmax activation as identity information enhancement to the outputs of every architecture and the formulation of the loss function will stay the same.

4.4 Extension to Swapped Face Videos

Although FACETRACER mainly functions at the image level, it can easily be extended to videos. For a suspect video V, we can extract source IDs from each frame or selected frames. This allows the inference process to revert to image-level extraction, where reference images I^r_{tar} or blank images can also assist in forensic analysis. Further details can be found in Section 5.6.

5 EXPERIMENT

5.1 Experimental Setting

Face-swapping Methods. We selected four widely-used face-swapping methods that explicitly disentangle identity and attribute information to separately train our FACETRACER, and demonstrate transferability among each other.

- **HiRes** [17] transfers different levels of attributes by three modules and learns a structure transfer direction in the latent space of StyleGAN [34], [35]. The face-swapping result is produced by swapping the identity-relevant latent codes of the target image and the refined latent codes corresponding to face attributes of the source image.
- **FaceShifter** [18] is an occlusion-aware face-swapping method, which first uses an adaptive embedding integration network (AEI-Net) to generate the first-stage swapped face image, followed by a heuristic error acknowledging refinement network (HEAR-Net) to produce better face-swapping result with occlusions.
- **SimSwap** [19] is based on an Encoder-Decoder architecture. It first utilize an encoder to extract attributes feature from the source image and uses the ID Injection Module (IIM) to transfer the identity information from the target face into the extracted attributes feature. After that, the decoder restores the modified features to the result swapped face image.
- **InfoSwap** [20] aims to disentangle identity-related information from facial images and then swap this information between different images while preserving other attributes. It also employs an encoder-decoder architecture as SimSwap to reconstruct images from the feature. Besides, InfoSwap applies an information bottleneck to the architecture, forcing the encoder to learn a compact representation that retains only the most relevant information for identity swapping and disentangle identity-related features from other attributes.

To further assess the transferability of FACETRACER on methods that do not explicitly decouple identity and attribute information, we also tested the model using swapped face images from the following two methods:

- **MegaFS** [21] produces face-swapping in the latent space of StyleGAN [34], [35] without extracting any identity or attribute information. The swapped face result was directly generated by the StyleGAN2 generator with the manipulated latent codes.
- **DiffSwap** [22] is a face-swapping method based on diffusion models [36], [37], which first extract identity information from the target image and inject the identity information into the conditional reverse diffusion process of the source face. Noted that the attribute information is not explicitly extracted during this process.

Moreover, we evaluated FACETRACER on two commercial face swapping apps, *i.e.*, Faceover [23] and DeepFaker [24].

Datasets. For each face-swapping method, the source and target face images are randomly selected from a pool of 30,000 identities (IDs) within the CelebA-HQ dataset [51]. To train FACETRACER, both swapped face images and raw face images are required. As outlined in the **Note** of Section 3.4, there is no overlap in identities between the training and inference phases. Specifically, IDs numbered 1 through 28,000 are allocated for the training set, while the remaining IDs are reserved for the testing set, and the identity pool in our experiment is built up by the identity information extracted by FACETRACER from these IDs. It is important to note that the number of images varies across different identities.

TABLE 1
Details of dataset construction used in FACETRACER.

Method	Format	Dataset	#Samples
HiRes [17]	Image	Train	113284+27816
		Test	2350+100
FaceShifter [18]	Image	Train	184985+27816
		Test	2675+100
SimSwap [19]	Image	Train	273875+27816
		Test	3163+100
InfoSwap [20]	Image	Train	271115+27816
		Test	3246+100
MegaFS [21]	Image	Test	2294+100
DiffSwap [22]	Image	Test	2493+100
FF++ [52]	Video	Test	2000+1000

While most contemporary face-swapping methods primarily operate at the image level as mentioned in Section 2.1, we also evaluate FACETRACER's performance on swapped face video with the most commonly-used dataset FF++ [52] that contains videos generated from four face-swapping methods: DeepFakes [53], FaceSwap [4], Face2Face [54], and NeuralTexture [55]. In contrast to some other video forgery datasets [56], [57], [58], FF++ labels the identity of the source and target person used in face-swapping rather than just the binary label of real or fake. Moreover, none of the identities in the FF++ dataset overlap with those in the CelebA-HQ dataset. Therefore, evaluation on FF++ can demonstrate transferablity of FACETRACER.

Table 1 provides a detailed breakdown of the dataset construction for FACETRACER, showing how each face-swapping method contributes to the dataset. In Samples column, we list the number of swapped content+raw content. As we mentioned in Section 3.3, FACETRACER should extract the correct identity information from un-swapped faces, thus we add some raw content into the test set, which are selected from those identities that produced the swapped face images in the test set.

The Baseline and Evaluation Metrics. We utilized the most commonly used identity information extractor, ArcFace [44], with a backbone of ResNet18 trained on MS1M [59] dataset as our baseline **B**. Since FACETRACER could be treated as a face identification system, we use the **Top-*k* Accuracy (Top-*k* ACC)** as the evaluation metric. This metric denotes the rate at which the correct label is among the top k labels predicted (ranked by similarity scores) by the face identification model. A higher Top-k ACC means that the extracted identity information can more effectively trace the source person.

Implementation Details. We have implemented FACETRACER on Pytorch platform and trained the model with a single NVIDIA A6000 GPU. Each input images and reference images are cropped and aligned to the size of 112×112 with the pre-trained MTCNN [38] model in both training and inference phases. The settings of the identity

TABLE 2
Performance comparison between the baseline and FACETRACER under different scenarios.

Method	Metric(%, ↑)	B	S1	S2	S3
HiRes [17]	Top-1 ACC	0.04	98.80	99.27	93.34
	Top-5 ACC	0.21	99.27	99.61	97.31
	Top-10 ACC	0.46	99.31	99.61	97.73
FaceShifter [18]	Top-1 ACC	0.03	99.40	99.40	99.40
	Top-5 ACC	0.22	99.55	99.55	99.55
	Top-10 ACC	0.41	99.55	99.55	99.55
SimSwap [19]	Top-1 ACC	0.03	99.13	99.19	98.73
	Top-5 ACC	0.40	99.41	99.47	99.16
	Top-10 ACC	0.70	99.41	99.50	99.25
InfoSwap [20]	Top-1 ACC	0.09	97.97	98.19	91.83
	Top-5 ACC	0.31	98.70	98.79	96.36
	Top-10 ACC	0.63	98.79	98.89	97.34

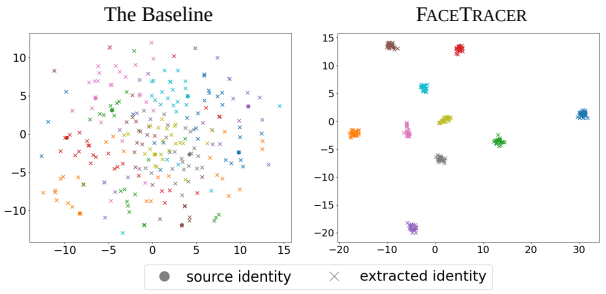


Fig. 7. T-SNE visualization of the ground-truth identity information of the source person and the identity information extracted by the baseline (left) and FACETRACER (right).

information extraction model and the loss function were listed in Section 4.1 and Section 4.2. During the training phase, we use the Ranger optimizer [60] with the parameters $(0.95, 0.999)$ and the Cosine learning rate scheduler to update the parameters of the model for 20 epochs, with an initialized learning rate of 1e-4, a weight decay of 2e-5, and a batch size of 64. The total training process of FACETRACER takes around 24 to 28 hours, depending on the magnitude of the data in the training set.

5.2 Overall Evaluation

In this part, we evaluated the effectiveness of FACETRACER in extracting identity information of the source person in swapped face images. As mentioned in Section 3.4, we set up three different scenarios, namely, the full-reference scenario (S1), the half-reference scenario (S2), and the none-reference scenario (S3). Here, the training set and inference set are built on the same face swapping methods.

As shown in Table 2, FACETRACER achieves an high top-1 accuracy under different scenarios, while the top-10 accuracy of the baseline is no more than 1%. This demonstrates that FACETRACER can **effectively** trace the source person in suspicious images with a high accuracy, regardless of whether the suspicious image has undergone face-swapping, while the baseline method fails in all cases. We explain that the baseline tends to extract the identity information relevant to the target person in the image rather than the source person. In contrast, FACETRACER mitigates the influence of the target person's identity information, enabling the accurate extraction of the identity of the source person. Noticeably, there is negligible difference between

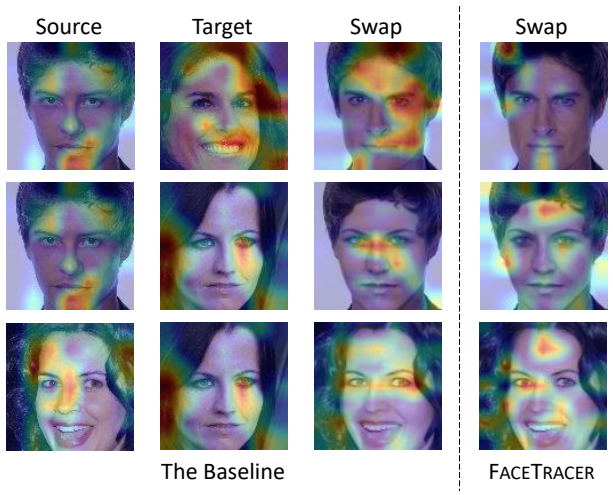


Fig. 8. Saliency map visualization of the images during the face-swapping process from the Baseline (left) and FACETRACER (right). FACETRACER focuses on regions such as hair, face shape, cheeks, bridge and forehead to extract the identity information of the source person.

S1 and S2, but both outperform S3, indicating that using reference images during the training phase can enhance FACETRACER's performance.

To better depict the effectiveness of our framework, we presented the t-SNE analysis results of the baseline and FACETRACER in Figure 7. We demonstrate this result using FACETRACER trained on the SimSwap method with the S1 setting as an example. Initially, we randomly selected 10 facial images of different people as the source faces and applied the SimSwap method to these images with different target faces which are also randomly selected. Subsequently, we extracted the identity information from these swapped face images with both the baseline method and FACETRACER. The results indicate that FACETRACER can accurately extract the identity information of the source individual from the input images, whereas traditional identity information extraction networks fail.

To better understand FACETRACER's ability to trace source identities, we use saliency maps to visualize what region the model focuses on. As illustrated in Figure 8, when processing swapped face images, the baseline model tends to focus on regions that are similar to those of the target image. This bias leads the baseline to identify these images as the target person, which is the training objective of the face swapping method. In contrast, FACETRACER effectively detects regions that are more relevant to recognizing the source identity, thereby enabling the extraction of source-related identity information for accurate tracing. Please note that FACETRACER places greater emphasis on regions such as hair, face shape, cheeks, bridge and forehead, these regions contain identity information of the source person even after face swapping algorithms are applied, as these regions are slightly changed after the face swapping process.

Additionally, we shall point out that it takes less than 0.05 seconds for FACETRACER to extract identity information of the source person from the input image, which shows the efficiency of FACETRACER.

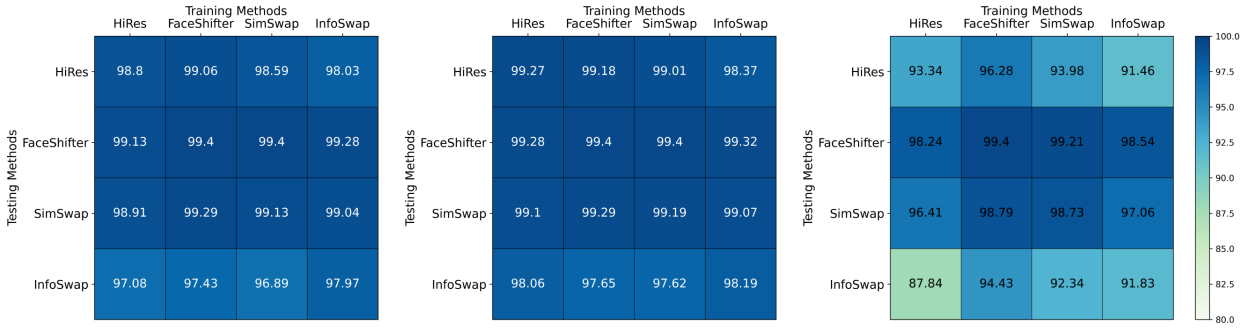


Fig. 9. Transferability of FACETRACER to different face-swapping methods that **explicitly** disentangle identity and attribute information, different figures represent different scenarios. Left: full-reference scenario (S1); Middle: half-reference scenario (S2); Right: none-reference scenario (S3). We adopt Top-1 ACC (↑) as the metrics. FACETRACER exhibits strong transferability across these methods.

Fig. 10. Transferability of FACETRACER to face-swapping methods that **implicitly** disentangle identity and attribute information, different figures represent different scenarios. Left: Full-reference scenario (S1); Middle: Half-reference scenario (S2); Right: None-reference scenario (S3). We adopt Top-1 ACC (↑) as the metrics. FACETRACER still exhibits transferability across these methods.

5.3 Transferability

In this part, we first evaluated the transferability of FACETRACER across four face-swapping methods that explicitly disentangle identity and attribute information, where each model was trained on one face-swapping method and tested on the other three face-swapping methods. We conducted evaluations under S1, S2, and S3 scenarios. As shown in Figure 9, FACETRACER holds strong transferability across these four face-swapping methods. Furthermore, the transferability evaluations conducted on FACETRACER show that using reference images during the training phase can enhance FACETRACER's transferability to some extent. For instance, when tested on swapped face images generated by InfoSwap [20], FACETRACER trained on HiRes under S1 and S2 scenarios achieve 97.08% and 98.06% Top-1 ACC, while the FACETRACER trained under S3 scenario only obtains 87.84% Top-1 ACC.

Moreover, we also conducted experiments on the transferability of FACETRACER in handling swapped face images generated by methods that do not explicitly disentangle identity and attribute information. Figure 10 shows that the performance of FACETRACER decreases on the swapped face images generated by these methods compared with the previous four face-swapping methods, but still maintains an acceptable level. Although these two methods do not explicitly disentangle identity and attribute information, the identity information of the source person still partially remains in the swapped face images. For instance, MegaFS [21] uses the deep StyleGAN latent codes of the source image as the latent codes for generating swapped face images, and DiffSwap [22] uses the noise image obtained by adding noise to the source image as the starting point for generating swapped face images. Therefore, FACETRACER still exhibits transferability to these methods. Additionally, we also observed an interesting point: when testing the transferability performance on MegaFS, there was a greater decrease in performance compared to other methods. This may be because MegaFS directly manipulates latent codes without extracting identity or attribute information. Figure 9 and Figure 10 only provide the Top-1 ACC results, and more results (Top-5&Top-10) can be seen in the supplementary material, which share a consistent conclusion.

We conducted experiment to simulate the situation of using FACETRACER without knowledge of the face swapping method, we randomly selected 1,000 images generated by the aforementioned six face swapping methods and tested the performance of FACETRACER trained on HiRes data, which achieved 92.78% Top-1 accuracy under S1 scenario.

Commercial Apps. For better demonstration of the performance of FACETRACER, we generate 50 swapped-face images with two commercial face swapping apps, Faceover [23] and DeepFaker [24], and FACETRACER demonstrates 100% and 96% tracing accuracy in average, respectively.

5.4 Intra- & Inter-gender Performance

In this part, we measured the performance of FACETRACER when dealing with different source and target genders.

TABLE 3
Performance comparison of FACETRACER under Intra- & Inter-gender face swapping (source←target)

Train	Metric(%, ↑)	Test(S1/S2/S3) All	M←M	F←F	M←F	F←M
HiRes [17]	Top-1 ACC	98.80/99.27/93.34	98.57/99.40/92.26	99.09/99.24/95.22	98.75/98.75/86.25	97.11/99.03/83.65
	Top-5 ACC	99.27/99.61/97.31	98.92/99.64/97.38	99.62/99.69/98.03	98.75/98.75/95.00	98.07/99.03/89.42
	Top-10 ACC	99.31/99.61/97.73	99.04/99.64/97.61	99.62/99.69/98.40	98.75/98.75/96.25	98.07/99.03/91.34
FaceShifter [18]	Top-1 ACC	99.40/99.40/99.40	99.65/99.65/99.65	99.40/99.40/99.40	98.95/98.95/98.95	98.07/98.07/98.07
	Top-5 ACC	99.55/99.55/99.55	99.65/99.65/99.65	99.66/99.66/99.66	98.95/98.95/98.95	98.07/98.07/98.07
	Top-10 ACC	99.55/99.55/99.55	99.65/99.65/99.65	99.66/99.66/99.66	98.95/98.95/98.95	98.07/98.07/98.07
SimSwap [19]	Top-1 ACC	99.13/99.19/98.73	99.79/99.79/99.42	99.03/99.09/98.79	98.71/99.03/96.62	97.97/97.97/98.07
	Top-5 ACC	99.41/99.47/99.16	99.79/99.79/99.58	99.45/99.57/99.33	99.35/99.35/97.29	97.97/97.97/98.71
	Top-10 ACC	99.41/99.50/99.25	99.79/99.79/99.58	99.45/99.63/99.45	99.35/99.35/97.63	97.97/97.97/98.71
InfoSwap [20]	Top-1 ACC	97.97/98.19/91.83	97.83/98.14/92.25	98.12/98.37/92.18	97.97/97.97/88.21	97.50/97.50/91.55
	Top-5 ACC	98.70/98.79/96.36	98.34/98.45/95.76	99.00/99.06/96.62	98.64/98.98/95.71	98.21/98.21/97.29
	Top-10 ACC	98.79/98.89/97.34	98.45/98.55/96.79	99.06/99.18/97.56	98.98/98.98/97.14	98.21/98.21/97.97

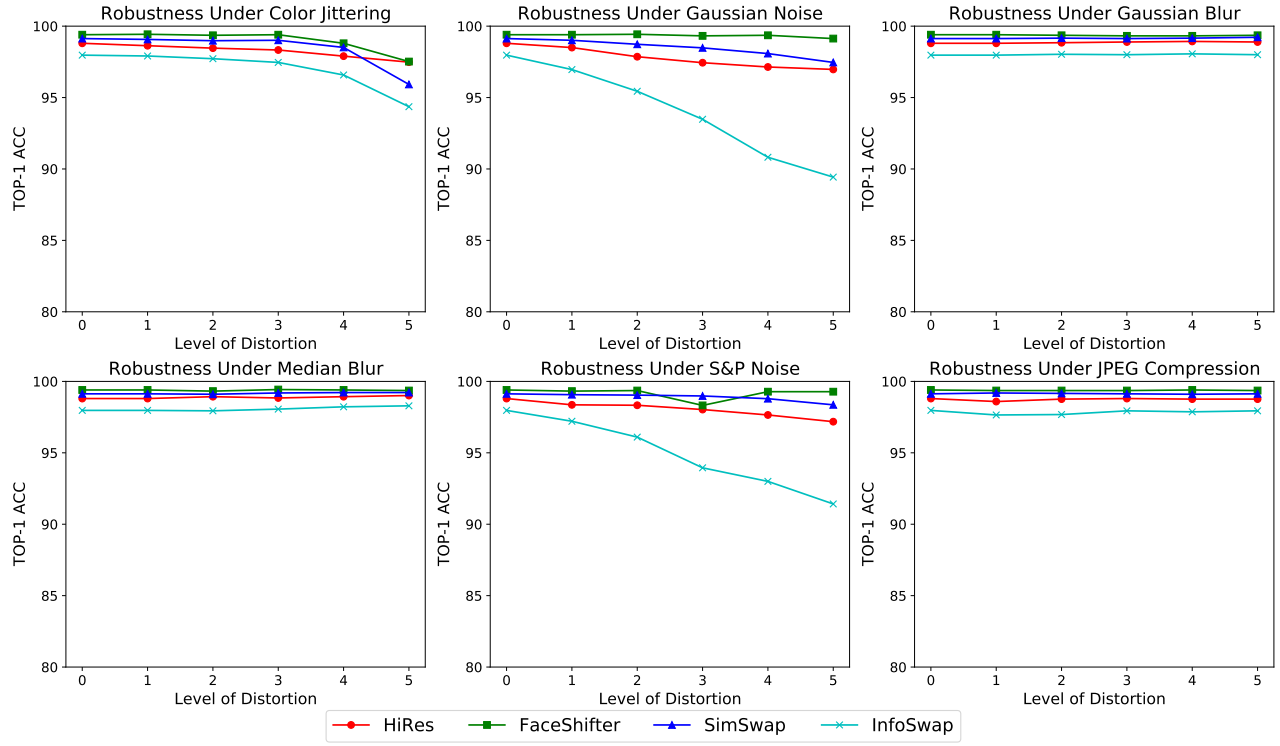


Fig. 11. FACETRACER holds excellent robustness under various distortions that may occur during the network transmission.

According to our experience, the selection of source and target people can affect the quality of swapped face images. Therefore, we divided the testing sets of each face-swapping method into four parts and tested the face identification performance of FACETRACER under different scenarios. Here, M←F indicates that the source person is male and the target person is female, the other notations follow the same logic. The results are listed in Table 3. It can be observed that FACETRACER does not lose face identification performance in almost every subsets under different scenarios, although some visually unfavorable swapped face images may be contained in the subsets.

5.5 Robustness

In this part, we investigated the robustness of FACETRACER, *i.e.*, its ability to handle various distortions that may occur during network transmission. We applied some common distortions to the testing images, including Gaussian noise, salt-and-pepper noise, Gaussian blur, median blur, JPEG compression, and color jittering. For each type of distortion, we set five levels to study the impact of increasing distortion on the performance of FACETRACER. We listed the parameters of different levels of distortions in the supplemetary material. Taking S1 scenario as an example (see Figure 11), FACETRACER exhibits outstanding robustness when dealing with Gaussian blur, median blur, and JPEG compression. Additionally, although the performance of FACETRACER decreases when facing the distortions like Gaussian noise, salt-and-pepper noise, and color jittering, it is still acceptable even under extreme distortions.

5.6 Extension to Swapped Face Videos

Here, we conducted experiments on the FF++ dataset [52] under the S1 scenario, and the results are demonstrated in Figure 12, where the frame count refers to the number of frames selected from the suspected video input V. Specifically, we randomly selected several frames from the video, and then fed the aligned images paired with the reference image $\{I^r_{tar}\}$ into FACETRACER, where $\{I^r_{tar}\}$ will be copied several times and thus paired with the selected

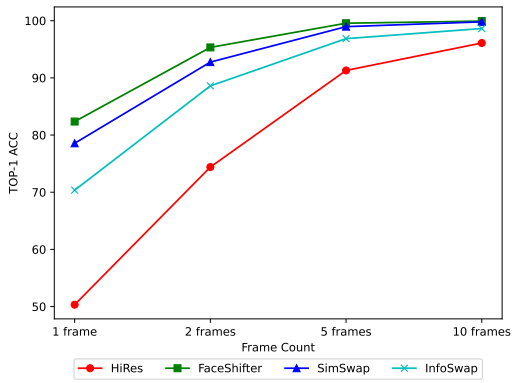


Fig. 12. Performance of FACETRACER when extents to swapped face videos under S1 scenario. Although FACETRACER performs not very effective with 1 frame input due to the video quality of FF++, its performance improves significantly as the number of input frames increases. We adopt Top-1 ACC (↑) as the metrics.

TABLE 4
Performance comparison of FACETRACER with different structures.

Method	Metric(%, ↑)	M1	M2	M3	M4
HiRes [17]	Top-1 ACC	98.80	78.71	97.56	89.51
	Top-5 ACC	99.27	84.51	98.67	95.17
	Top-10 ACC	99.31	87.24	98.89	96.37
FaceShifter [18]	Top-1 ACC	99.40	99.25	99.28	98.05
	Top-5 ACC	99.55	99.48	99.47	99.02
	Top-10 ACC	99.55	99.48	99.51	99.21
SimSwap [19]	Top-1 ACC	99.13	99.23	99.16	98.39
	Top-5 ACC	99.41	99.44	99.47	99.22
	Top-10 ACC	99.41	99.48	99.47	99.38
InfoSwap [20]	Top-1 ACC	97.97	97.94	97.91	96.64
	Top-5 ACC	98.70	98.54	98.79	98.10
	Top-10 ACC	98.79	98.64	98.92	98.35

video frames. The final identity information could be simply computed as the average of the several outputs.

It can be observed that increasing the number of input video frames effectively improves the performance of FACETRACER. More results could be seen in the supplementary material. This will greatly facilitates the real-world scenario, where evidence for fraud and information about the target person is more likely to appear in video format, and the defender will be able to extract more accurate identity information about the attacker. Additionally, it is noted that FACETRACER exhibits not-so-good transferability performance when testing on the FF++ dataset. This could be due to the low resolution of FF++ itself, and the fact that the facial region occupies only a portion of the entire video. As a result, additional interpolation operations are required for face alignment to achieve a resolution of 112×112, leading to its not-so-good performance.

5.7 Ablation Studies

Model Structure. In this part, we first performed ablation studies among different structures of FACETRACER:

- M1: Default setting of FACETRACER.
- M2: Replacing the ViT-S structure with ResNet18 structure used in the identity information extraction module and keeping the rest part of FACETRACER unchanged.
- M3: Replacing the disentanglement module with a cross-attention module and keeping the rest part of FACETRACER unchanged.

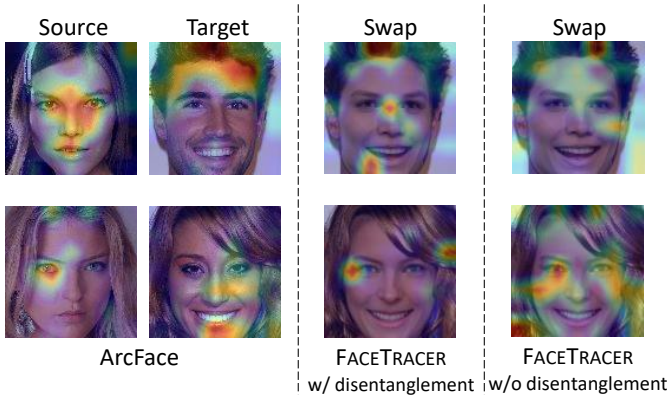


Fig. 13. Saliency map visualization of FACETRACER with and without the identity information disentanglement module. The identity information disentanglement module effectively shifts FACETRACER's focus from regions analogous to the target face to those resembling the source face.

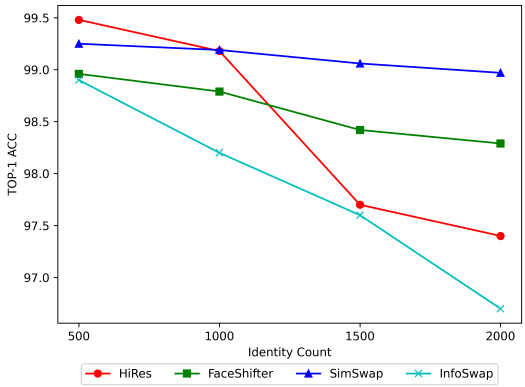


Fig. 14. Performance of FACETRACER when the size of identity pool varies. FACETRACER maintains consistent performance as the size of identity pool becomes larger. We adopt Top-1 ACC (↑) as the metrics.

- M4: Replacing the identity information enhancement module with a plain softmax activation as Eq. (2) and keeping the rest part of FACETRACER unchanged.

Under S1 scenario, we can observe from Table 4 that different model architectures (M1, M2 and M3) do not significantly affect the performance of FACETRACER, while changing the identity information enhancement module (M1 and M4) will affect FACETRACER's performance under each training set. This demonstrates the flexibility of FACETRACER's model structure and the effectiveness of the identity information enhancement module.

To demonstrate the effectiveness of the identity information disentanglement module, we used saliency maps to illustrate regions that FACETRACER focuses on with and without this module. As shown in Figure 13, the identity information disentanglement module successfully enables the region FACETRACER focuses on switching from being similar to the target face to being similar to the source face.

The Scale of the Identity Pool. We evaluated FACETRACER's performance across varying identity pool sizes of 500, 1,000, 1,500, and 2,000 identities. For each configuration, we conducted analyses on distinct test sets, each comprising 1,000 randomly selected swapped face images, where the source identities were constrained to the respective identity pools.

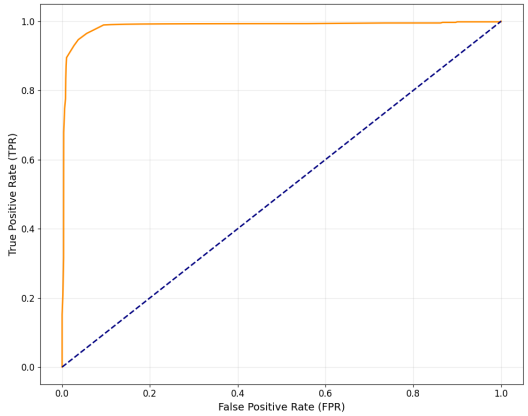


Fig. 15. Receiver Operating Characteristic (ROC) curve of the filtering strategy, this filtering strategy is simple yet effective.

TABLE 5
Performance of FACETRACER trained on the multiple face-swapping dataset.

Testing Method	Metric(%, ↑)	S1	S2	S3
HiRes [17]	Top-1 ACC	99.23	99.36	93.38
	Top-5 ACC	99.61	99.65	97.14
	Top-10 ACC	99.61	99.65	97.90
FaceShifter [18]	Top-1 ACC	99.36	99.32	98.50
	Top-5 ACC	99.51	99.55	99.21
	Top-10 ACC	99.55	99.58	99.40
SimSwap [19]	Top-1 ACC	99.22	99.25	97.03
	Top-5 ACC	99.50	99.53	98.70
	Top-10 ACC	99.50	99.53	99.07
InfoSwap [20]	Top-1 ACC	98.16	98.67	90.79
	Top-5 ACC	98.63	98.92	96.13
	Top-10 ACC	98.73	98.95	97.18

Figure 14 illustrates that FACETRACER maintains consistent performance as the number of identities in the identity pool increases. Additionally, since FACETRACER is intended for use by trusted third parties—such as police forces and other organizations capable of maintaining large-scale identity pools—we can assume that the source identity information of input images already exists within these pools. To further optimize performance, we can initially filter out images whose source identities are not present in the identity pool by applying an identity matching threshold. We evaluated our filtering strategy through Receiver Operating Characteristic (ROC) curve analysis, presented in Figure 15. The performance characteristics, plotted across multiple filtering threshold configurations, demonstrate the effectiveness of the strategy.

5.8 Ensemble Training Strategy

We trained FACETRACER on dataset built on a single face swapping method by default. Here, we further evaluated the impact of ensemble training strategy, namely training on dataset consisting of different face-swapping methods. Specifically, we sampled one-fourth of the swapped face images from the four training sets respectively and made a training dataset with multiple face-swapping methods. Then we trained FACETRACER on this dataset. As shown in Table 5, FACETRACER achieves excellent performance on different testing sets under different scenarios. Therefore, users of FACETRACER could collect training data from different face-swapping methods for training without affecting FACETRACER's performance.

TABLE 6
Performance of FACETRACER against continuous face swapping.

Method	Metric(%,↑)	S1	S2	S3
HiRes [17]	Top-1 ACC	97.85	98.56	84.73
	Top-5 ACC	98.75	99.23	92.41
	Top-10 ACC	99.09	99.28	93.79
FaceShifter [18]	Top-1 ACC	100.0	100.0	99.59
	Top-5 ACC	100.0	100.0	99.89
	Top-10 ACC	100.0	100.0	99.94
SimSwap [19]	Top-1 ACC	99.33	99.23	94.69
	Top-5 ACC	99.63	99.74	97.55
	Top-10 ACC	99.74	99.79	98.16
InfoSwap [20]	Top-1 ACC	95.95	96.40	90.09
	Top-5 ACC	97.77	97.87	93.19
	Top-10 ACC	98.27	98.27	97.65

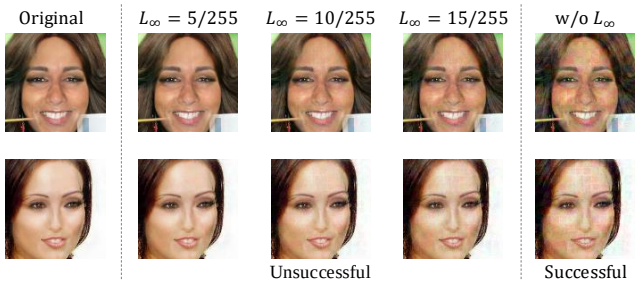


Fig. 16. Visual examples of adaptive evaluation. A successful attack, however, only occurs without L_∞ constraints, sacrificing visual quality with clearly perceptible noise.

5.9 Continuous Face-swapping Attacks

In this part, we considered adversaries who perform face-swapping on a face image with two different target faces. For example, Alice is the malicous attacker, she uses Bob's face image as the target face image to produce swapped face image. Then, Carol's face image is further used as the second target face image to continuously produce the swapped face image, both by the same face-swapping method. In this scenario, FACETRACER needs to be able to extract Alice's identity information. In Table 6, FACETRACER still achieves high accuracy on double-swapped face images, which demonstrates FACETRACER could effectively defense such adversary. In the supplementary material, we also demonstrated the ability of FACETRACER to defend against an adaptive adversary using two different face-swapping methods with two different target faces.

5.10 Adaptive Evaluation

In this part, we considered adaptive adversaries who can access and obtain the output of FACETRACER. One intuitive adaptive attack is to circumvent FACETRACER's tracing by adding imperceptible adversarial noise to swapped face images. The goal of this attack is to alter the identity information extracted by FACETRACER so that it no longer matches the source person. To generate such adversarial noise, we utilized the widely recognized PGD [61] method to generate adversarial noise under different L_∞ constraints. As shown in Figure 16, the attack fails when $L_\infty = 15/255$, where the added noise becomes clearly visible. We also present a case where the attack succeeds, but at the cost of severely compromising visual quality, making it impractical for attackers in real-world scenarios.

6 Related Works

Face Swapping Techniques. Early face-swapping methods [25], [55], [62], [63] rely on 3D templates to disentangle identity and attribute information to process face-swapping. However, these methods lack expressiveness for some detailed features such as illumination and style. Recently, many work introduce GANs for face swapping, such as RSGAN [64], FSNet [65] and FSGAN [66]. The encoder-decoder architecture is also commonly used in various face-swapping methods [18], [19], [20], which explicitly disentangles the identity of the target person and the attributes of the source person by encoding them separately with different encoders, and then uses a decoder to obtain the swapped face image. With the development of the generative models, some face-swapping methods also leverage the generative capability of the off-the-shelf models. MegaFS [21] and HiRes [17] leverage the generative capabilities of StyleGAN [34], [35] to achieve face swapping, while DiffSwap [22] utilizes state-of-the-art generative model, namely the diffusion models [36], [37] to accomplish face swapping.

Face Swapping Detection. As the antithesis of face-swapping, face-swapping detection technology is also constantly evolving. Early works [67], [68] detect the forgery through visual biological artifacts. Some work focus on the frequency domain of the swapped face images and videos and produced methods such as [69], [70] while some others focus on the temporal consistency such as [11], [12], [71]. Moreover, recent approach also captured precise face geometric features [72] or blending artifacts [9] to detect face-swapping images and videos. Some recent work [73], [74], [75] uses face identity information to perform face swapping detection, and [10] has also found implicit identities for face-swapping results, which has somewhat inspired our approach.

7 Discussion

Although FaceTracer performs well as shown in the above experiments, there also exist some limitations.

Quality of Swapped Face Images. Since the input image size of FaceTracer is fixed, some low-resolution swapped face images may have facial parts smaller than the required input size and need scaling operations. This can lead to less accurate extraction of the source person's identity information, *i.e.*, for low-quality swapped face data like FF++ [52], FaceTracer performs unsatisfactorily with single frame input. A highly effective solution to this problem is to increase the number of input images, which can significantly enhance FaceTracer's performance (see Figure 12).

Face Image Reconstruction. Although FaceTracer extracts the identity information of the source individual, a more convenient tracing method is reconstructing the attacker's face directly from the extracted identity information using facial reconstruction techniques. This can establish an end-to-end system. We attempted to construct such a facial reconstruction network on the CelebA-HQ dataset. Unfortunately, in some attempts, this reconstruction network lacked the generalization ability to out-of-domain data. We will investigate this in future work.

Face Swapping Methods. In Section 5.3, we found that FaceTracer exhibits a decrease in performance on swapped face images generated by MegaFS and DiffSwap. These methods, which do not explicitly disentangle identity and attribute information, retain less identity information of the source person. Nevertheless, these methods often come with drawbacks. For instance, MegaFS may easily generate meaningless images, and DiffSwap requires significant computational resources for training. In practical scenarios, traditional face-swapping methods that explicitly disentangle identity and attribute information are still predominant. Furthermore, as of now, there isn't a method that can entirely separate identity from attribute information, allowing the extraction of identity information from the source person to remain feasible.

Privacy Enhancing Methods. As discussed, FaceTracer could leverage privacy-preserving technologies to safeguard privacy and prevent the leakage of identity information. Fortunately, a variety of techniques are available for achieving this with large-scale facial datasets. These techniques include differential privacy [76], [77], feature subtraction [78], and adding adversarial noise [48]. Moreover, using edge computing-based method could also be a possible option [79]. More details can refer to the latest survey [80].

8 Conclusion

In this paper, we proposed the first non-intrusive tracing framework, FaceTracer, which can extract the identity information of the source person in swapped face images for effective forensics. To achieve it, we elaborate two main modules: identity information extraction module and identity information disentanglement module. Extensive qualitative and quantitative results demonstrate the effectiveness of FaceTracer under three practical scenarios and its strong transferability and robustness. FaceTracer also exhibits the flexibility in model structure and can be easily extended to face-swapping videos. In the future, we consider building an end-to-end framework to reconstruct face images of the source person from swapped face images.

References

[1] "Snapchat," https://www.snapchat.com/lens/dc6a7589a13f49eea647591ab428bb67.
[2] "Faceapp," https://www.faceapp.com/.
[3] "Deepfacelab," https://github.com/iperov/DeepFaceLab.
[4] "Faceswap," https://github.com/deepfakes/faceswap.
[5] "Deepfacelive," https://github.com/iperov/DeepFaceLive.
[6] Foxnews, "How yahoo boys use real time face swapping to carry out elaborate romance scams," https://www.foxnews.com/tech/how-yahoo-boys-use-real-time-face-swapping-to-carry-out-elaborate-romance-scams.
[7] ABC, "How south-east asia's pig butchering scammers are using artificial intelligence technology," https://www.abc.net.au/news/2024-05-16/pig-butchering-scams-artificial-intelligence-ai-face-swapping-/103804830.
[8] CNN, "Finance worker pays out $25 million after video call with deepfake 'chief financial officer'," https://edition.cnn.com/2024/02/04/asia/deepfake-cfo-scam-hong-kong-intl-hnk/index.html.
[9] K. Shiohara and T. Yamasaki, "Detecting deepfakes with self-blended images," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 18 720–18 729.

[10] S. Dong, J. Wang, R. Ji, J. Liang, H. Fan, and Z. Ge, "Implicit identity leakage: The stumbling block to improving deepfake detection generalization," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 3994–4004.

[11] Z. Wang, J. Bao, W. Zhou, W. Wang, and H. Li, "Altfreezing for more general video face forgery detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 4129–4138.

[12] Y. Xu, J. Liang, G. Jia, Z. Yang, Y. Zhang, and R. He, "Tall: Thumbnail layout for deepfake video detection," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 22 658–22 668.

[13] Y. Zhang, D. Ye, C. Xie, L. Tang, X. Liao, Z. Liu, C. Chen, and J. Deng, "Dual defense: Adversarial, traceable, and invisible robust watermarking against face swapping," *IEEE Transactions on Information Forensics and Security*, vol. 19, pp. 4628–4641, 2024.

[14] P. Sun, H. Qi, Y. Li, and S. Lyu, "Faketracer: Catching face-swap deepfakes via implanting traces in training," 2024.

[15] Z. Xinghui, Z. Wenbo, W. Tianyi, C. Shen, Y. Taiping, D. Shouhong, Z. Weiming, and Y. Nenghai, "Rank-based no-reference quality assessment for face swapping," *arXiv preprint arXiv:2406.01884*, 2024.

[16] B. Huang, Z. Wang, J. Yang, J. Ai, Q. Zou, Q. Wang, and D. Ye, "Implicit identity driven deepfake face swapping detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2023, pp. 4490–4499.

[17] Y. Xu, B. Deng, J. Wang, Y. Jing, J. Pan, and S. He, "High-resolution face swapping via latent semantics disentanglement," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2022, pp. 7642–7651.

[18] L. Li, J. Bao, H. Yang, D. Chen, and F. Wen, "Faceshifter: Towards high fidelity and occlusion aware face swapping," *arXiv preprint arXiv:1912.13457*, 2019.

[19] R. Chen, X. Chen, B. Ni, and Y. Ge, "Simswap: An efficient framework for high fidelity face swapping," in *Proceedings of the 28th ACM international conference on multimedia*, 2020, pp. 2003–2011.

[20] G. Gao, H. Huang, C. Fu, Z. Li, and R. He, "Information bottleneck disentanglement for identity swapping," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 3404–3413.

[21] Y. Zhu, Q. Li, J. Wang, C.-Z. Xu, and Z. Sun, "One shot face swapping on megapixels," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 4834–4844.

[22] W. Zhao, Y. Rao, W. Shi, Z. Liu, J. Zhou, and J. Lu, "Diffswap: High-fidelity and controllable face swapping via 3d-aware masked diffusion," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 8568–8577.

[23] "Faceover," https://apps.apple.com/us/app/faceover-photo-face-swap/id393476155.

[24] "Deepfaker," https://apps.apple.com/us/app/face-swap-video-by-deep-fake/id1625343601.

[25] H. Kim, P. Garrido, A. Tewari, W. Xu, J. Thies, M. Niessner, P. Pérez, C. Richardt, M. Zollhöfer, and C. Theobalt, "Deep video portraits," *ACM transactions on graphics (TOG)*, vol. 37, no. 4, pp. 1–14, 2018.

[26] K. Nagano, J. Seo, J. Xing, L. Wei, Z. Li, S. Saito, A. Agarwal, J. Fursund, H. Li, R. Roberts *et al.*, "pagan: real-time avatars using dynamic textures." *ACM Trans. Graph.*, vol. 37, no. 6, p. 258, 2018.

[27] T. Li, T. Bolkart, M. J. Black, H. Li, and J. Romero, "Learning a model of facial shape and expression from 4d scans." *ACM Trans. Graph.*, vol. 36, no. 6, pp. 194–1, 2017.

[28] Y. Deng, J. Yang, S. Xu, D. Chen, Y. Jia, and X. Tong, "Accurate 3d face reconstruction with weakly-supervised learning: From single image to image set," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, 2019, pp. 0–0.

[29] Y. Feng, H. Feng, M. J. Black, and T. Bolkart, "Learning an animatable detailed 3D face model from in-the-wild images," vol. 40, no. 8, 2021. [Online]. Available: https://doi.org/10.1145/3450626.3459936

[30] R. Danecek, M. J. Black, and T. Bolkart, "EMOCA: Emotion driven monocular face capture and animation," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022, pp. 20 311–20 322.

[31] J. Kim, J. Lee, and B.-T. Zhang, "Smooth-swap: A simple enhancement for face-swapping with smoothness," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2022, pp. 10 779–10 788.

[32] Y. Wang, X. Chen, J. Zhu, W. Chu, Y. Tai, C. Wang, J. Li, Y. Wu, F. Huang, and R. Ji, "Hififace: 3d shape and semantic prior guided high fidelity face swapping," 2021.

[33] K. Preechakul, N. Chatthee, S. Wizadwongsa, and S. Suwajanakorn, "Diffusion autoencoders: Toward a meaningful and decodable representation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 10 619–10 629.

[34] T. Karras, S. Laine, and T. Aila, "A style-based generator architecture for generative adversarial networks," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 4401–4410.

[35] T. Karras, S. Laine, M. Aittala, J. Hellsten, J. Lehtinen, and T. Aila, "Analyzing and improving the image quality of stylegan," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 8110–8119.

[36] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," *Advances in neural information processing systems*, vol. 33, pp. 6840–6851, 2020.

[37] J. Song, C. Meng, and S. Ermon, "Denoising diffusion implicit models," *arXiv preprint arXiv:2010.02502*, 2020.

[38] 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, pp. 1499–1503, 2016.

[39] J. Wang, Y. Yuan, and G. Yu, "Face attention network: An effective face detector for the occluded faces," *arXiv preprint arXiv:1711.07246*, 2017.

[40] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[41] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly *et al.*, "An image is worth 16x16 words: Transformers for image recognition at scale," *arXiv preprint arXiv:2010.11929*, 2020.

[42] W. Liu, Y. Wen, Z. Yu, M. Li, B. Raj, and L. Song, "Sphereface: Deep hypersphere embedding for face recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 212–220.

[43] H. Wang, Y. Wang, Z. Zhou, X. Ji, D. Gong, J. Zhou, Z. Li, and W. Liu, "Cosface: Large margin cosine loss for deep face recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 5265–5274.

[44] J. Deng, J. Guo, N. Xue, and S. Zafeiriou, "Arcface: Additive angular margin loss for deep face recognition," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 4690–4699.

[45] M. Kim, A. K. Jain, and X. Liu, "Adaface: Quality adaptive margin for face recognition," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 18 750–18 759.

[46] W. Xia, Y. Yang, J.-H. Xue, and B. Wu, "Tedigan: Text-guided diverse face image generation and manipulation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 2256–2265.

[47] O. Patashnik, Z. Wu, E. Shechtman, D. Cohen-Or, and D. Lischinski, "Styleclip: Text-driven manipulation of stylegan imagery," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 2085–2094.

[48] Z. Wang, H. Wang, S. Jin, W. Zhang, J. Hu, Y. Wang, P. Sun, W. Yuan, K. Liu, and K. Ren, "Privacy-preserving adversarial facial features," 2023. [Online]. Available: https://arxiv.org/abs/2305.05391

[49] H. Phan, C. X. Le, V. Le, Y. He, and A. ". Nguyen, "Fast and interpretable face identification for out-of-distribution data using vision transformers," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, January 2024, pp. 6301–6311.

[50] J. Deng, Y. Chen, Y. Zhong, Q. Miao, X. Gong, and W. Xu, "Catch you and i can: Revealing source voiceprint against voice conversion," in *32nd USENIX Security Symposium (USENIX Security 23)*, 2023, pp. 5163–5180.

[51] C.-H. Lee, Z. Liu, L. Wu, and P. Luo, "Maskgan: Towards diverse and interactive facial image manipulation," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[52] A. Rossler, D. Cozzolino, L. Verdoliva, C. Riess, J. Thies, and M. Nießner, "Faceforensics++: Learning to detect manipulated facial images," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, pp. 1–11.

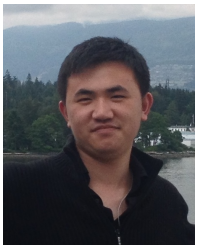
[53] "Deepfakes," https://github.com/ondyari/FaceForensics/tree/master/dataset/DeepFakes.
[54] J. Thies, M. Zollhöfer, M. Stamminger, C. Theobalt, and M. Nießner, "Face2face: Real-time face capture and reenactment of rgb videos," 2020.
[55] J. Thies, M. Zollhöfer, and M. Nießner, "Deferred neural rendering: Image synthesis using neural textures," *Acm Transactions on Graphics (TOG)*, vol. 38, no. 4, pp. 1–12, 2019.
[56] B. Dolhansky, J. Bitton, B. Pflaum, J. Lu, R. Howes, M. Wang, and C. C. Ferrer, "The deepfake detection challenge (dfdc) dataset," 2020.
[57] L. Yuezun, Y. Xin, S. Pu, Q. Honggang, and L. Siwei, "Celeb-df: A large-scale challenging dataset for deepfake forensics," in *IEEE Conference on Computer Vision and Patten Recognition (CVPR)*, 2020.
[58] L. Jiang, R. Li, W. Wu, C. Qian, and C. C. Loy, "DeeperForensics-1.0: A large-scale dataset for real-world face forgery detection," in *CVPR*, 2020.
[59] Y. Guo, L. Zhang, Y. Hu, X. He, and J. Gao, "Ms-celeb-1m: A dataset and benchmark for large-scale face recognition," 2016.
[60] L. Wright, "Ranger - a synergistic optimizer." https://github.com/lessw2020/Ranger-Deep-Learning-Optimizer, 2019.
[61] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu, "Towards deep learning models resistant to adversarial attacks," 2019. [Online]. Available: https://arxiv.org/abs/1706.06083
[62] K. Olszewski, Z. Li, C. Yang, Y. Zhou, R. Yu, Z. Huang, S. Xiang, S. Saito, P. Kohli, and H. Li, "Realistic dynamic facial textures from a single image using gans," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 5429–5438.
[63] Y. Nirkin, I. Masi, A. T. Tuan, T. Hassner, and G. Medioni, "On face segmentation, face swapping, and face perception," in *2018 13th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2018)*. IEEE, 2018, pp. 98–105.
[64] R. Natsume, T. Yatagawa, and S. Morishima, "Rsgan: face swapping and editing using face and hair representation in latent spaces," *arXiv preprint arXiv:1804.03447*, 2018.
[65] ——, "Fsnet: An identity-aware generative model for image-based face swapping," in *Computer Vision–ACCV 2018: 14th Asian Conference on Computer Vision, Perth, Australia, December 2–6, 2018, Revised Selected Papers, Part VI 14*. Springer, 2019, pp. 117–132.
[66] Y. Nirkin, Y. Keller, and T. Hassner, "Fsgan: Subject agnostic face swapping and reenactment," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, pp. 7184–7193.
[67] C. M. Liy and L. InIctuOculi, "Exposingaicreated fakevideos-bydetectingeyeblinking," in *2018IEEEInterG national Workshop on Information Forensics and Security (WIFS). IEEE*, 2018.
[68] X. Yang, Y. Li, and S. Lyu, "Exposing deep fakes using inconsistent head poses," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 8261–8265.
[69] H. Liu, X. Li, W. Zhou, Y. Chen, Y. He, H. Xue, W. Zhang, and N. Yu, "Spatial-phase shallow learning: Rethinking face forgery detection in frequency domain," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2021, pp. 772–781.
[70] J. Wei, S. Wang, and Q. Huang, "F^3net: fusion, feedback and focus for salient object detection," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 34, no. 07, 2020, pp. 12 321–12 328.
[71] Y. Zheng, J. Bao, D. Chen, M. Zeng, and F. Wen, "Exploring temporal coherence for more general video face forgery detection," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2021, pp. 15 044–15 054.
[72] L. Li, J. Bao, T. Zhang, H. Yang, D. Chen, F. Wen, and B. Guo, "Face x-ray for more general face forgery detection," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 5001–5010.
[73] X. Dong, J. Bao, D. Chen, W. Zhang, N. Yu, D. Chen, F. Wen, and B. Guo, "Identity-driven deepfake detection," 2020. [Online]. Available: https://arxiv.org/abs/2012.03930
[74] B. Liu, B. Liu, M. Ding, T. Zhu, and X. Yu, "Ti2net: temporal identity inconsistency network for deepfake detection," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2023, pp. 4691–4700.
[75] B. Fan, Z. Jiang, S. Hu, and F. Ding, "Attacking identity semantics in deepfakes via deep feature fusion," in *2023 IEEE 6th International Conference on Multimedia Information Processing and Retrieval (MIPR)*. IEEE, 2023, pp. 114–119.
[76] M. Chamikara, P. Bertok, I. Khalil, D. Liu, and S. Camtepe, "Privacy preserving face recognition utilizing differential privacy," *Computers & Security*, vol. 97, p. 101951, Oct. 2020. [Online]. Available: http://dx.doi.org/10.1016/j.cose.2020.101951
[77] Y. Mao, S. Yi, Q. Li, J. Feng, F. Xu, and S. Zhong, "A Privacy-Preserving deep learning approach for face recognition with edge computing," in *USENIX Workshop on Hot Topics in Edge Computing (HotEdge 18)*. Boston, MA: USENIX Association, Jul. 2018. [Online]. Available: https://www.usenix.org/conference/hotedge18/presentation/mao
[78] Y. Mi, Z. Zhong, Y. Huang, J. Ji, J. Xu, J. Wang, S. Wang, S. Ding, and S. Zhou, "Privacy-preserving face recognition using trainable feature subtraction," 2024. [Online]. Available: https://arxiv.org/abs/2403.12457
[79] X. Wang, H. Xue, X. Liu, and Q. Pei, "A privacy-preserving edge computation-based face verification system for user authentication," *IEEE Access*, vol. 7, pp. 14 186–14 197, 2019.
[80] L. Laishram, M. Shaheryar, J. T. Lee, and S. K. Jung, "Toward a privacy-preserving face recognition system: A survey of leakages and solutions," *ACM Comput. Surv.*, jun 2024, just Accepted. [Online]. Available: https://doi.org/10.1145/3673224

Zhongyi Zhang is currently working toward the PhD degree in School of Cyber Science and Technology, University of Science and Technology of China (USTC). His research interests mainly include Face Anonymization, DeepFake Detection and AIGC Generation.

Jie Zhang received his B.S. degree in 2017 from China University of Geosciences, Beijing and received his Ph.D.degree in 2022 from the University of Science and Technology of China (USTC). Currently, he is a Research Scientist of Center for Frontier AI Research, Agency for Science, Technology and Research (A*STAR), Singapore. His primary research interests include IP protection for AI, Trustworthy generative AI, and AI Regulation.

Wenbo Zhou received the BS degree from Nanjing University of Aeronautics and Astronautics, China in 2014, and the PhD degree from the University of Science and Technology of China in 2019, where he is currently vise professor. His research interests include information hiding and AI security.

Xinghui Zhou is a PhD student at the University of Science and Technology of China in Anhui, China. He obtained his bachelor's degree in Automation Engineering from Tianjin University and his master's degree in Computing Science from Beijing Electronic Science & Technology Institute, where he focused on Visual Computing, Image Quality Assessment, and Artificial Intelligence.



Qing Guo received his Ph.D. degree from the School of Computer Science and Technology, Tianjin University, China. He was a research fellow and the Wallenberg-NTU Presidential Postdoctoral Fellow at the Nanyang Technological University, Singapore, from Dec. 2019 to Sep. 2022. He is currently a senior research scientist and principal investigator at the Center for Frontier AI Research (CFAR), A*STAR in Singapore. He is also an adjunct assistant professor at the National University of Singapore (NUS). He serves as the Senior PC for AAAI and Area Chair for ICLR 2025. His research mainly focuses on computer vision, AI security, adversarial attacks, and robustness. He is a member of IEEE.

Weiming Zhang received the MS and PhD degrees from Zhengzhou Information Science and Technology Institute, China in 2002 and 2005 respectively. Currently, he is a professor with the School of Information Science and Technology, University of Science and Technology of China. His research interests include information hiding and multimedia security.

Tianwei Zhang is an associate professor of College of Computing and Data Science at Nanyang Technological University. His research focuses on computer system security. He is particularly interested in security threats and defenses in machine learning systems, autonomous systems, computer architecture and distributed systems. He received his Bachelor's degree at Peking University in 2011, and the Ph.D. degree in at Princeton University in 2017.

Nenghai Yu received the PhD degree from USTC in 2004. He is a full professor with the University of Science and Technology of China. He is also the director of Information Processing Center of USTC, deputy director of academic committee of School of Information Science and Technology. He was a visiting scholar in Institute of Production Technology, Faculty of Engineering, University of Tokyo, in 1999 and did cooperative research as the senior visiting scholar in Department of Electrical Engineering, Columbia University, from Apr. to Oct. 2008. His research focuses on image processing and video analysis, multimedia communication, media content security, Internet information retrieval, data mining and content filtering, network communication and security.