

Stealthy Physical Masked Face Recognition Attack via Adversarial Style Optimization

Huihui Gong^{ID}, Minjing Dong^{ID}, Siqi Ma^{ID}, Seyit Camtepe^{ID}, Senior Member, IEEE,
Surya Nepal^{ID}, Senior Member, IEEE, and Chang Xu^{ID}, Senior Member, IEEE

Abstract—Deep neural networks (DNNs) have achieved state-of-the-art performance on face recognition (FR) tasks in the last decade. In real scenarios, the deployment of DNNs requires taking various face accessories into consideration, like glasses, hats, and masks. In the COVID-19 pandemic era, wearing face masks is one of the most effective ways to defend against the novel coronavirus. However, DNNs are known to be vulnerable to adversarial examples with a small but elaborated perturbation. Thus, a facial mask with adversarial perturbations may pose a great threat to the widely used deep learning-based FR models. In this paper, we consider a challenging adversarial setting: targeted attack against FR models. We propose a new stealthy physical masked FR attack via adversarial style optimization. Specifically, we train an adversarial style mask generator that hides adversarial perturbations inside style masks. Moreover, to ameliorate the phenomenon of sub-optimization with one fixed style, we propose to discover the optimal style given a target through style optimization in a continuous relaxation manner. We simultaneously optimize the generator and the style selection for generating strong and stealthy adversarial style masks. We evaluated the effectiveness and transferability of our proposed method via extensive white-box and black-box digital experiments. Furthermore, we also conducted physical attack experiments against local FR models and online platforms.

Index Terms—Adversarial attack, computer vision, deep learning, face recognition, physical setting.

I. INTRODUCTION

IN RECENT years, deep learning models have achieved great success in the face recognition (FR) task. A common deep FR model's working process is as follows: 1) utilizing convolutional neural networks (CNNs) to learn the face features of input face images; 2) constructing efficient loss function (like contrastive loss [1], [2], triplet loss [3] and center loss [4]) for distinguishing

Manuscript received 17 June 2023; revised 29 August 2023 and 20 October 2023; accepted 21 October 2023. Date of publication 3 November 2023; date of current version 21 March 2024. This work was supported by the Australian Research Council under Grants DP210101859 and FT230100549. The Associate Editor coordinating the review of this manuscript and approving it for publication was Prof. Qianqian Xu. (*Corresponding author: Chang Xu.*)

Huihui Gong, Minjing Dong, and Chang Xu are with the School of Computer Science, Faculty of Engineering, University of Sydney, Sydney, NSW 2008, Australia (e-mail: hgong9611@uni.sydney.edu.au; mdong0736@uni.sydney.edu.au; c.xu@sydney.edu.au).

Siqi Ma is with the School of Engineering and Information Technology, University of New South Wales Canberra, Canberra, ACT 2612, Australia (e-mail: sqiqi.ma@adfa.edu.au).

Seyit Camtepe and Surya Nepal are with Data61, CSIRO, Sydney, NSW 1466, Australia (e-mail: sayit.camtepe@data61.csiro.au; surya.nepal@data61.csiro.au).

Digital Object Identifier 10.1109/TMM.2023.3330089

the face features; 3) using the well-trained models by 1) and 2) as well as some verification face images to define the distance threshold of different identities; 4) recognizing whether the two testing face images are the same person. Such an FR model and its improved versions [3], [4], [5], [6], [7], [8] can achieve very high accuracy on popular face image datasets. Therefore, these FR models are widely used in daily work and life. Many companies use these models to differentiate their employees and outsiders. Besides, lots of public places deploy them to find out specific persons. These high-accuracy FR models improve our life efficiency and enhance public security.

Since the end of 2019, a novel coronavirus (COVID-19 coronavirus) has been sweeping the world. Its effect on every aspect of our lives and work will continue in the near future. A simple and effective way to protect us from this virus is to wear face masks when we go into public places. However, the widely used face masks pose a threat to FR models: diverse face masks reduce the recognition rate; and, worse still, attackers deliberately elaborate some adversarial face masks to deceive FR models.

Adversarial vulnerability is a prevalent problem in the computer vision field: attackers elaborate examples with adversarial perturbations that can mislead the deep learning models [9], [10], [11], [12]. Plenty of existing works have shown that adversarial examples (though with very small perturbations) can completely destroy the deep learning models [9], [10]. Particularly, adversarial attacks are an enormous menace to FR models: black-box adversarial examples were proposed against FR models in [13], [14]; adversarial face accessories were applied to attacking FR models (like glass, hat, and makeup) in [15], [16], [17]. However, most existing attacks achieve stealthiness by either crafting small perturbations or modifying semantic attributes of victim images. Worse still, these methods cannot achieve the satisfactory trade-off between attack strength and stealthiness. For example, PGD-based attack method [18] obtains good stealthiness but a low attack success rate when the perturbation limitation is small; while with bigger perturbations, its attack success rate will be higher, but the generated perturbations can be easily perceived by human. For the adversarial patch and AdvMask methods [19], [20], stealthiness is not a factor in their consideration. Besides, for the physical setting, the performance of the 2D attacks (like PGD and adversarial patch) will be compromised when projecting the generated 2D perturbations or patches into 3D physical space. In this work, we wonder whether we can generate targeted stealthy physical

adversarial perturbations against FR models in the physical setting. Style transfer is to transfer the texture of a source image to a target image while retaining the content of the source image. Recent works [21], [22] attempt to hide adversarial perturbations in the process of transferring styles, which generate more natural adversarial examples in their setting.

In this paper, we propose a novel stealthy adversarial style mask to perform targeted attacks against FR models in both digital and physical settings. Concretely, we train an adversarial style mask generator that hides adversarial perturbations inside the style masks. However, a fixed style usually incurs sub-optimal performance. To alleviate this phenomenon, we introduce a continuous relaxation on style via a weighted sum of style sets to discover the optimal style for a particular target. We optimize the style selection and the generator simultaneously for generating stealthy adversarial style masks (SASMask). Next, the SASMask is used to attack FR models in the digital setting and the physical setting. To better maintain the aggressivity in the physical world, we also adopt the adaptation mapping/transformation to robustize our SASMask. Before ending this section, we list the main contributions of this paper below:

- We propose a novel stealthy adversarial style mask that hides adversarial perturbations inside the diverse style face masks via adversarial style optimization to attack FR models.
- We design to simultaneously train the adversarial style mask (SASMask) generator and optimize the style selection to alleviate the local minima problem and generate stronger adversarial masks.
- Extensive digital and physical experiments are conducted to demonstrate the effectiveness of our proposed SASMask in deceiving FR models.

II. RELATED WORK

A. Face Recognition

In recent decades, deep FR models have achieved great progress [3], [4], [5], [7], [12], [23], [24]. Existing deep FR models mostly utilize the idea of mapping a pair of face images to a distance proposed in [25]. In 2015, Ding and Tao [23] extracted complementary facial features from multimodal data to obtain robust face representation. Besides, Schroff et al. [3] proposed the effective triplet loss, which minimizes the distance between an anchor and a positive example and maximizes the distance between an anchor and a negative example until the margin is satisfied. Later, further works [4], [5], [6], [7] were proposed to improve metric learning: Wen et al. [4] proposed the center loss to learn a center for deep features of each class and penalize the distances between the deep features and their corresponding class centers; Liu et al. [5] proposed the angular softmax loss that enables to learn angularly discriminative features; Wang et al. [6] proposed the large margin cosine loss, and Deng et al. [7] improve it by adding the parameter m to the inside of cos function. Gao et al. [24] used adaptive neighbor embedding to learn robust face image super-resolution for better recognition.

B. Adversarial Attack

First introduced in [9], the adversarial attack is to generate samples (i.e., adversarial examples) with imperceptible perturbations that can mislead deep models. From the attack scenario, the adversarial attack can be utilized in a digital setting and in a physical setting.

Digital Attack: Numerous previous works show that deep models are vulnerable to adversarial examples in many fields, like classification [26], [27], natural language processing [28], object detection [29], speech recognition [30], etc. These attacks can be either restricted or unrestricted. The strong 1st-order Projected Gradient Descent (PGD) [18] is a famous work of restricted attacks. Besides, unrestricted adversarial examples are also extensively studied, which try to alter some significant attributes or components of images, like color [31], [32], texture [33], [34] as well as modifications of partial areas of images [19], [35]. Notwithstanding, such malicious samples are usually unnatural or distorted. Additionally, unrestricted attacks cannot mount adversarial examples with complicated patterns, which limits their usage in practical situations. Gao et al. [36] proposed an effective transferable digital adversarial attack by adding perturbation to clean images based on the attended regions and features. Wan et al. [37] presented a strong white-box digital adversarial attack based on average gradient.

Physical Attack: Kurakin et al. [38] showed that the digital adversarial examples are still effective in subverting deep neural models by printing and recapturing them with a camera. However, the attack performance plummets because of viewpoint shifts, camera noise, or other physical transformations [39]. As such, more robust physical attacks demand larger perturbations and adaptations towards physical transformations. Expectation Over Transformation (EoT) [39] is an effective technique to maintain the attack performance in the physical setting. Other robust physical attacks include adversarial patch (AdvPatch) [19], malicious graffiti [40], adversarial laser beams [41], adversarial clothes [42], adversarial shadows [43] and so forth. Most of these physical attacks generate large perturbations to increase the attack strength, which inevitably accounts for large and unnatural distortions of original images. Moreover, widely used FR models attract much attention to revealing the potential threats against them. Generally, such threats are divided into the dodging (untargeted) attack and the impersonation (targeted) attack. The dodging attack tries to decrease the similarity of same-identity pairs, while the impersonation attack aims to increase the similarity between the test face and the target identity, which is more difficult than the untargeted attack technically. Thus, we mainly focus on the impersonation face attack in this paper. There are some related works of this task: adversarial glasses [15], adversarial hat [16], eye patch [44], adversarial makeup [45]. Besides, Zolfi et al. [20] proposed an adversarial mask (AdvMask) that is quite related to our work. They chose a basic pattern and optimized it, which starts training the adversarial mask from a manually selected pattern, while our method not only optimizes the perturbation generation but also optimizes the style weights for selecting optimal basic patterns. Besides, AdvMask only uses the total variation loss as the constraint loss, which

is not enough to constrain the stealthiness of the generated perturbations, whereas we propose explicit losses to constrain the magnitude of perturbations.

C. Style Transfer

Style transfer is to transfer the texture of a source image to a target image while retaining the content of the source image. With the help of deep neural networks, neural methods [46], [47], [48] achieved remarkable progress for the style transfer task, where the content and style features are learned by CNNs. Then, the style features of the style image are combined into the target image to realize style transfer. Recently, a few works [21], [22] generated adversarial examples with style transfer. In this paper, we hide adversarial perturbations in the masks to generate natural physical attacks against FR models. Different from existing works, we propose the adversarial attack in multiple styles and optimize the combination weights of these styles.

III. METHODOLOGY

The attack success rate (ASR) of an adversarial attack is always regarded as one of the most important criteria, however, the naturalness of generated physical adversarial examples in real scenarios becomes another necessary factor in the evaluation [21], [22]. In this work, we focus on the targeted adversarial attack on physical masks via different styles. Previous style-transfer-based physical attacks aim at untargeted attacks with a specific style. However, these techniques cannot be directly applied in our setting since we consider a more complex targeted attack. With a fixed style, it could be difficult for attackers to generate targeted physical adversarial masks with better trade-offs between ASR and naturalness. We argue that different styles lead to different trade-offs for a target. Thus, instead of utilizing a fixed style for style-transfer attack which could result in sub-optimal solutions, we propose to search for the optimal style for each target in adversarial mask generation.

A. Problem Formulation

Given a face image $x \in \mathbb{R}^m$ with its ground truth identity y , a face recognition model v maps the image to a feature embedding e , as well as a predicted identity. Given a target identity $y_t \neq y$, the goal of a targeted adversarial attack on the facial mask is to discover a natural mask with perturbation M^{adv} which fools the model v to predict the target identity. Formally, the adversarial mask is defined as

$$\min_{M^{adv}} \mathcal{L}_{FR}(v(R(x, M^{adv})), e_t),$$

where $e_t = v(x_t)$, (1)

where \mathcal{L}_{FR} is the face recognition loss, e_t denotes the feature embedding of target y_t , and $R(x, M^{adv})$ is the overlay operation that overlays adversarial mask M^{adv} on face x . A traditional solution to (1) can be an adversarial attack via an additive noise as $M^{adv} = M + \delta$ where M is the original mask and δ denotes the perturbation. Although the masks generated by adversarial attacks can be aggressive, those masks are always far away from

Algorithm 1: Training Stage of SASMask.

Data: face images x , an original mask pattern c , style set S , trained style transfer network G_{ori} , targeted face image x_t , victim face recognition model v , training epoch N

Result: SASMask generator G_{adv} , optimized style weights w

```

1 Initialize  $G_{adv}^0 = G_{ori}$ ,  $w^0 = [0]_K$ ;
2 Generate original style mask  $G_{ori}(c, \tilde{h}^0 \cdot S)$ ;
3 Compute targeted embedding  $e_t = v(x_t)$ ;
4 for  $n = 1, \dots, N$  do
5   Get SASMask by  $G_{adv}(c, \tilde{h}^0 \cdot S)$ ;
6   Get the mask position map of SASMask by [49]
    and overlay it on face images to obtain
     $R(x, G_{adv}(x, \tilde{h}^0 \cdot S))$  ;
7   Randomly transform the masked faces
     $T(R(x, G_{adv}(c, \tilde{h}^0 \cdot S)))$  ;
8   Calculate the loss in Eq. 9;
9   Separately compute generator gradient  $\nabla_{G_{adv}^n} \mathcal{L}$ 
    and weight gradient  $\nabla_{w^n} \mathcal{L}$ ;
10  Update  $G_{adv}^{n+1}$  with  $G_{adv}^n$  and  $\nabla_{G_{adv}^n} \mathcal{L}$  by some
    optimizer (like ADAM [50]);
11  Update  $w^{n+1}$  with  $w^n$  and  $\nabla_{w^n} \mathcal{L}$  by some
    optimizer;
12 end

```

Algorithm 2: Inference Stage of SASMask.

Data: Validation set \mathcal{V} , test face images x , style set S , SASMask generator G_{adv}^* , targeted face image x_t , victim face recognition model v , optimized style weights w^*

Result: ASR, SSIM

- 1: Compute targeted embedding $e_t = v(x_t)$;
- 2: Compute recognition threshold κ via Validation set \mathcal{V} ;
- 3: Select the optimized adversarial mask style
 $s^* = \text{argmax } w^* \cdot S$;
- 4: Get SASMask by $G_{adv}^*(c, s^*)$;
- 5: Get the mask position map of SASMask by [49] and
 overlay it on face images to obtain $R(x, G_{adv}^*(x, s^*))$;
- 6: Calculate ASR via Eq. (10);
- 7: Calculate SSIM via Eq. (11);

those natural ones in the real world, which makes the attack perceptible in practice, such as PGD and AdvPatch [18], [19].

B. Stealthy Adversarial Style Mask

Instead of directly applying adversarial attacks, we propose to incorporate attacks into a customized style. Specifically, we utilize the style transfer techniques to generate stealthy adversarial masks. Given the content of the original mask pattern c and the style of predefined style image s , the adversarial mask M^{adv} in (1) can be generated via a style-transferred manner as

$$M^{adv} = G_{adv}(c, s). \quad (2)$$

where G_{adv} denotes the adversarial style transfer generator. Similarly, we denote the one without attack as G_{ori} . In order to guarantee stealthiness of M^{adv} , we include l_1 -norm loss \mathcal{L}_1 to reduce the perturbation magnitude of image pixels, a total variation (TV) loss \mathcal{L}_{tv} to reduce the variance of image pixels, a content loss \mathcal{L}_c to remain the content of the source pattern and a style loss \mathcal{L}_s to generate the customized style. Formally, given a pre-trained feature extractor $\phi(\cdot)$, these loss terms can be formulated as

$$\begin{aligned}\mathcal{L}_1 &= |M^{adv} - M| = |G_{adv}(c, s) - G_{ori}(c, s)|, \\ \mathcal{L}_{tv} &= \sum_{i,j} \sqrt{(M_{i,j}^{adv} - M_{i+1,j}^{adv})^2 + (M_{i,j}^{adv} - M_{i,j+1}^{adv})^2}, \\ \mathcal{L}_c &= \sum_{i \in l_C} \|\phi_i(M^{adv}) - \phi_i(c)\|_2^2, \\ \mathcal{L}_s &= \sum_{j \in l_S} \|\mathcal{G}(\phi_j(M^{adv})) - \mathcal{G}(\phi_j(s))\|_2^2,\end{aligned}\quad (3)$$

where $\phi_l(\cdot)$ is the output of the feature extractor at layer l , l_C and l_S are the sets of content and style layers; $\mathcal{G}(\phi_l(\cdot))$ is the Gram matrix associated with the output at layer l . Thus (1) can be reformulated as

$$\min_{G_{adv}} \mathcal{L}(v, R, G_{adv}, G_{ori}, \phi; x, e_t, c, s),$$

$$\text{where } \mathcal{L} = \mathcal{L}_{FR} + \lambda_1 \mathcal{L}_1 + \lambda_{tv} \mathcal{L}_{tv} + \lambda_c \mathcal{L}_c + \lambda_s \mathcal{L}_s, \quad (4)$$

where $\lambda_1, \lambda_{tv}, \lambda_c$, and λ_s denote the scaling hyperparameters.

Physical Adaptation: Since physical environments are not as ideal as digital conditions, they usually involve noises (like environmental noises and camera noises) as well as natural transformations (like viewpoint shifts and rotations) in a physical setting. Therefore, we need to use physical adaptations to handle such diverse conditions. Here, we use the end-to-end 3D face UV position map [49] as the overlay operation $R(x, M^{adv})$ to overlay facial masks on faces for generating vivid and natural physically adaptive facial masks. Besides, we also utilize an auxiliary method similar to Expectation Over Transformation (EoT) [39]. In our setting, we aim to improve the robustness of SASMask towards noises, rotations, and viewpoint shifts. Here, we use a transformation function $T(\cdot)$ to transform the masked face data as $T(R(x, M^{adv}))$.

C. Adversarial Style Optimization

Note that the stealthy adversarial style masks generated by (4) could achieve sub-optimal adversarial strength, because a fixed style is not always a good initial point for different targets. As shown in Fig. 5, when we randomly sample one style from the style set, the optimization stops at different points in which the attack performances vary greatly. We mainly attribute this instability to a sub-optimal solution via a fixed style. Thus, in order to perform stable effective targeted adversarial style masks, styles are incorporated into the optimization. Specifically, we propose to optimize the convex combination of adversarial styles from a style set. Given a style set S that contains K style candidates, we aim to discover a superior style that achieves better trade-offs

between attack success rate and stealthiness than other potential styles. The objective in (4) becomes a bi-level optimization which can be reformulated as

$$\begin{aligned}\min_{G_{adv}} \mathcal{L}(v, R, G_{adv}, G_{ori}, \phi; x, e_t, c, s^*), \\ \text{s.t. } s^* = \operatorname{argmin}_{s \in S} \mathcal{L}(v, R, G_{adv}, G_{ori}, \phi; x, e_t, c, s)\end{aligned}\quad (5)$$

Because style selection is a discrete problem as shown in (5), it is difficult to optimize it directly. Hence, we propose to conduct continuous relaxation on s via the incorporation of style weight as:

$$s = \mathbf{w} \cdot S = \sum_k^K w_k S_k. \quad (6)$$

where $\mathbf{w} = [w_1, \dots, w_K]$ is the weight vector of K styles so that the style s becomes a weighted sum of style set S . During the inference stage, the style with maximum \mathbf{w} is selected as the optimal style for adversarial style attack as

$$s^* = h \cdot S, \text{ where } h = \operatorname{one_hot}(\operatorname{argmax}_k w_k), \quad (7)$$

where h denotes the one-hot distribution. Since (7) is not differentiable in the training phase, we propose to use the softmax with a temperature parameter to approximate the one-hot distribution h in a differentiable manner as

$$\tilde{h}_i(w_i) = \frac{\exp(w_i)/\tau}{\sum_{j=1}^K \exp(w_j)/\tau}, \quad (8)$$

where $\tau > 0$ is the temperature parameter, used for controlling the concentration of style selection. By setting τ to a smaller value during the training phase (e.g., 0.1), \tilde{h}_i is close to one-hot distribution h , which alleviates the gap between the training and inference phases. The objective in (5) can be reformulated as

$$\begin{aligned}\min_{G_{adv}} \mathcal{L}(v, R, G_{adv}, G_{ori}, \phi; x, e_t, c, s^*), \\ \min_{\mathbf{w}} \mathcal{L}(v, R, G_{adv}, G_{ori}, \phi; x, e_t, c, \mathbf{w}, S), \\ \text{s.t. } s^* = \tilde{h}(\mathbf{w}) \cdot S.\end{aligned}\quad (9)$$

During the inference phase, \tilde{h} is replaced by the one-hot distribution h via argmax on \mathbf{w} to select the optimal style s^* given a target. Thus, with the involvement of style weights, an optimal style can be discovered via our framework for each target instead of the sub-optimal solutions with a fixed style in previous works. We further summarize our proposed algorithm in Fig. 1 and Algorithms 1 and 2.

IV. EXPERIMENTS

In this section, we evaluated the effectiveness of SASMask in the digital setting and the physical setting. Specifically, we first outlined the experimental setup, and then we displayed the white-box and black-box experimental results in the digital setting, including multiple targets, cross-backbone, cross-head, and cross-dataset experiments. Besides, we analyzed SASMask via the ablation study for adversarial style optimization and loss ablation. Eventually, we also conducted physical experiments:

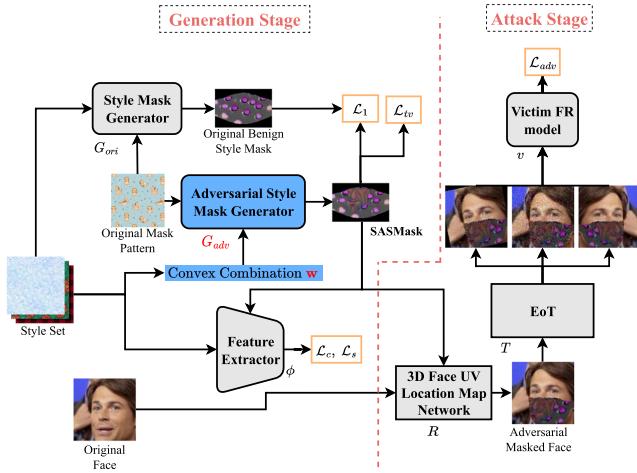


Fig. 1. Overview of our proposed method, which is divided into two stages: the generation stage and the attack stage. Parameters and networks that need to be optimized are emphasized in red bold font.

TABLE I
ARCHITECTURE OF (ADVERSARIAL) STYLE MASK GENERATOR

Index	Operation	Input Size	Output Size
1	Conv(in=3,out=32,k=9,p=(4,4,4),s=1)+InstanceNorm+ReLU	(3,112,112)	(32,112,112)
2	Conv(in=32,out=64,k=3,p=(0,1,0,s=2)+InstanceNorm+ReLU	(32,112,112)	(64,56,56)
3	Conv(in=64,out=128,k=3,p=(0,1,0,s=2)+InstanceNorm+ReLU	(64,56,56)	(128,28,28)
4	$\left[\text{Conv}(in=128,out=128,k=3,p=(1,1,1),s=1)+\text{InstanceNorm}+\text{ReLU}, \text{Conv}(in=128,out=128,k=3,p=(1,1,1),s=1)+\text{InstanceNorm}+\text{Residual Addition} \right] \times 5$	(128,28,28)	(128,28,28)
5	Interpolate(scale_factor=2) Conv(in=128,out=64,k=3,p=(1,1,1),s=1)+InstanceNorm+ReLU	(128,28,28)	(64,56,56)
6	Interpolate(scale_factor=2) Conv(in=64,out=32,k=3,p=(1,1,1),s=1)+InstanceNorm+ReLU	(64,56,56)	(32,112,112)
7	Conv(in=32,out=3,k=9,p=(4,4,4),s=1)+InstanceNorm+Sigmoid	(32,112,112)	(3,112,112)

offline printed experiments and online platform attack experiments.

A. Experimental Setup

a) *FR model and mask generator*: To fully evaluate our proposed method, we utilized five backbones and five state-of-the-art metric learning heads to train our SASMask generator and tested the trained model in the white-box and black-box settings. The five backbones are ResNet (–34, –50, –101) [51], MobileNet [52] and GhostNet [53]. Additionally, the heads are ArcFace [7], CosFace [6], CircleLoss [54], CurricularFace [55], MagFace [56]. For the (adversarial) style mask generator, we design a simplified U-net [57] like network to generate (adversarial) style masks, whose architecture is shown in Table I.

b) *Dataset*: We used four popular facial image datasets in our experiments: LFW [58], VGGFace2 [59], AgeDB [60], CFP [61]. We followed the setup of existing works, e.g., [13], [20], [45]. We randomly chose fifty identities from the LFW dataset as our targets for comprehensively evaluating the performance of our methods. In the training stage, we randomly selected 400 images of different identities (different from targeted identities) from the LFW dataset. In the testing stage, we randomly chose another 1000 images (different from targeted identities and training images) from the LFW dataset. Besides, we also randomly selected 1000 facial images from the other three datasets (VGGFace2, AgeDB, and CFP) for testing the

transferability to other datasets. For the training stage, subset data is enough, because there is little improvement with the full set, which needs much more computational resources.

c) *Baseline attack*: We compared our proposed method with several existing representative adversarial methods: Projected Gradient Descent (PGD) [18], Adversarial Patch (AdvPatch) [19], and AdvMask [20]. PGD is the strongest first-order digital attack; AdvPatch is one of the typical unrestricted attacks that can be directly used in the physical setting; AdvMask is a recently proposed adversarial mask. In addition, we also generated the randomly perturbed face mask as another baseline attack method. In addition to the above mask region-based comparisons, we also compared our method with other face attack methods: PGD on face (PGDFace) [18], AdvPatch (PatchFace) on face [44], AdvGlass [15], AdvHat [16], AdvMakeup [45]. For all the comparison methods, we follow their official experimental settings.

d) *Metric*: To quantify the attack performance of our method and the comparison methods, we used the *attack success rate* (ASR) as our attack metric, which is here defined as

$$\text{ASR} = \frac{\sum \mathbb{1}(d(\text{Test Face}, \text{Target}) < \kappa)}{\# \text{ of All Test Pairs}} \times 100\%, \quad (10)$$

where $d(\cdot, \cdot)$ is the distance metric; κ is the distance threshold and we used the K -fold cross-validation of the full test set to calculate it. Moreover, we also utilized the *structure similarity* (SSIM) [62] as our mask quality metric, whose definition is given by

$$\text{SSIM}(a, b) = \frac{(2\mu_a\mu_b + C_1) \cdot (2\sigma_{ab} + C_2)}{(\mu_a^2 + \mu_b^2 + C_1) \cdot (\sigma_a^2 + \sigma_b^2 + C_2)}, \quad (11)$$

where a and b are two masked facial images; μ_a (or μ_b) is the image mean; σ_a (or σ_b) is the image standard deviation; σ_{ab} denotes the covariance between a and b ; C_1 and C_2 are two constants ($C_1 = 0.0001$ and $C_2 = 0.0009$ in our experiments). Hereof, we computed the SSIM between the original style masked face image and the attacked masked image to evaluate the stealthiness of the deployed attack method, where larger SSIM means better stealthiness. Note that we used the value of SSIM to quantitatively calculate the metric of stealthiness (i.e., naturalness, the quality or state of being natural).

e) *Implementation detail*: We first pre-trained a benign style transfer generator G_{ori} on the Microsoft COCO dataset [63]. Then, we used this generator to initialize the SASMask generator and fine-tuned it with the Adam optimizer [50]. We set the initial model learning rate as 0.01 and the weight learning rate as 0.01. The values of λ_1 , λ_{tv} , λ_c and λ_s were set to 100, 10, 0.001 and 1000, respectively. Note that we utilized the FR heads and the classification idea to train the FR backbones; in the testing stage, we only used the trained FR backbones to obtain the feature embeddings for verification. Besides, all the victim face recognition models are trained on the CASIA-WebFace dataset [64]. The temperature τ was set to 0.1. For the input format, we set the size of input masked face images to be [112,112] and set the size of masks to be [60,112] in the bottom area of face images. We set the starting optimized points of different attack methods to be the original style masks. For the feature extractor,



Fig. 2. Chosen original mask pattern and ten style candidates.

TABLE II
TRADE-OFF (ASR AND SSIM) WITH HYPERPARAMETER TUNING

Tuning	λ_1	λ_{tv}	λ_c	λ_s	τ	ASR	SSIM
-	100	10	0.001	1000	0.1	98.8%	0.9130
λ_1	10	10	0.001	1000	0.1	99.1%	0.8732
	1000	10	0.001	1000	0.1	0.0%	0.9265
λ_{tv}	100	1	0.001	1000	0.1	98.8%	0.9066
	100	100	0.001	1000	0.1	0.0%	0.9220
λ_c	100	10	0.0001	1000	0.1	98.9%	0.9037
	100	10	0.01	1000	0.1	97.6%	0.9146
λ_s	100	10	0.001	100	0.1	98.3%	0.9134
	100	10	0.001	10000	0.1	98.4%	0.8993
τ	100	10	0.001	1000	1	82.3%	0.8467
	100	10	0.001	1000	0.01	96.8%	0.7530

The best ASR and SSIM results are stressed in bold.

we use the VGG-16 architecture [65], where we use the ReLU outputs of layers 2, 4, 7, and 10 to compute the content and style losses. For the end-to-end 3D face UV location map network, we use the well-trained PRNet [49]. For the batch normalization of (masked) facial images, we use the mean of (0.5,0.5,0.5) and the standard deviation of (0.5,0.5,0.5); for the batch normalization of VGG-16 inputs, we use the mean of (0.485, 0.456, 0.406) and the standard deviation of (0.229, 0.224, 0.225). By observing common fashionable mask patterns on the market, we select ten patterns to construct a pre-defined style set as our style candidates. Besides, we choose an original mask pattern as the style transfer initial point to generate stylized mask patterns with the pre-defined style set. The chosen original pattern and the style candidates are shown in Fig. 2.

f) Hyperparameter tuning: To tune loss scaling hyperparameters (λ_1 , λ_{tv} , λ_c and λ_s), we first ran the model without optimization and obtain every loss, then initialized the hyperparameters to make them the same order of magnitude, and used the variable-controlling method to tune hyperparameters for better performance. To tune temperature τ , we first selected several values (0.001, 0.01, 0.1, 1, 10) to run our algorithm, and then based on the optimized results, we initialized τ as the corresponding value (0.1) with the best performance. Some specific tuning results are shown in Table II. Besides, we further used the early stopping trick [66] that stops the training when there is no improvement after seven epochs for quicker convergence and better performance. Normally, the training process takes about 50 epochs (about 3 hours) with a single 2080Ti GPU. The convergence curves are shown in Fig. 3.

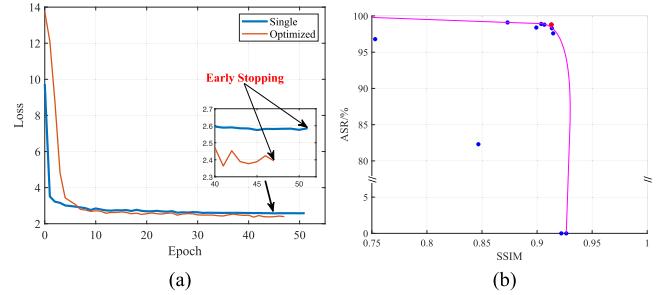


Fig. 3. Convergence curves of optimized and single adversarial style masks as well as a scatter plot of hyperparameter tuning. (a) Convergence curves of optimized and single adversarial style masks. (b) Scatter plot of hyperparameter tuning, where the magenta curve is the approximate Pareto curve and the red point is near the Pareto front, where the corresponding hyperparameters are set as the final training hyperparameters.

B. DIGITAL ATTACK

1) White-Box Attack

We first evaluated our proposed method in a white-box setting. Specifically, we randomly selected four identities from the LFW dataset as the targets and trained corresponding SASMask generators. Here, we used the ResNet-50 backbone and the ArcFace head. Furthermore, we also tested the baseline attack method, i.e., random perturbations, PGD, AdvPatch, and AdvMask. For PGD attack methods, the maximum perturbations were set to be 64/255, 80/255, and 128/255 with 40 iteration steps; For AdvPatch attack methods, the radii of the patches were set to be 20, 24, and 28 (a larger radius would be beyond the range of the mask) with 100 iterations. For the sake of fairness, it is noted that we set the total variation loss in AdvMask and the stealthiness loss to a similar order of magnitude. The experimental results are shown in Table III , from which we can find that our SASMask can deceive the FR model with the best ASR compared to the baseline attacks, reducing the recognition rate from about one hundred percent up to about zero percent with the highest SSIMs. Besides, as for the baseline attacks, randomly perturbed masks and AdvPatch attacked masks (even though with very large perturbation, like radius = 28) rarely work towards adversarially attacking the FR model. Some adversarial masked face examples can be seen in Fig. 4.

To demonstrate the effectiveness of the proposed method, we further conducted multi-target attack experiments with up to 50 targets. The results are shown in Table IV, from which we can see as the number of targets increases, ASR and SSIM of our method stabilize at around 90.0% and 0.9170, respectively. However, the counterparts are ineffective or unstable as the number of targets increases. Through statistics, the optimization selections of different styles of the 50 targets are: Style 0 was chosen three times; Style 1 was chosen eight times; Style 2 was chosen four times; Style 3 was chosen six times; Style 4 was chosen two times; Style 5 was chosen four times; Style 6 was chosen five times; Style 7 was chosen six times; Style 8 was chosen five times; Style 9 was chosen seven times. From the results, we can see that no one style is clearly dominant. The style optimization selection is related to both the style pattern and the attack target.

TABLE III
ASR AND SSIM OF DIFFERENT ATTACK METHODS WITH DIFFERENT HYPERPARAMETERS OF FOUR DIFFERENT TARGETS

Target	Attack Method	Original Face	Rand	PGD-64	PGD-80	PGD-128	AdvPatch-20	AdvPatch-24	AdvPatch-28	AdvMask	Ours
Vivica Fox (Style 1)	ASR	0.1%	0.3%	48.9%	49.6%	51.1%	3.9%	4.9%	5.1%	98.6%	98.8%
	SSIM	-	0.7166	0.8998	0.8858	0.8545	0.8265	0.7964	0.7590	0.6614	0.9130
Patricia Hearst (Style 8)	ASR	0.7%	2.7%	9.1%	10.1%	12.3%	6.6%	6.8%	7.9%	81.7%	85.3%
	SSIM	-	0.7290	0.9001	0.8979	0.8963	0.7160	0.6580	0.5868	0.5926	0.9122
Aaron Eckhart (Style 5)	ASR	0.0%	0.1%	3.6%	8.4%	30.5%	0.0%	0.0%	0.1%	89.3%	91.7%
	SSIM	-	0.7418	0.8541	0.8445	0.7871	0.8359	0.8089	0.7920	0.7683	0.8844
Steve Park (Style 9)	ASR	0.9%	1.3%	9.4%	8.2%	10.6%	3.3%	4.6%	5.3%	78.5%	84.4%
	SSIM	-	0.7591	0.9065	0.8969	0.8887	0.7852	0.7026	0.6222	0.6082	0.9181

Here, we study the ResNet-50 backbone and the arcface head as the victim FR model. The best results are stressed in bold. The optimized style in our proposed method is written below the target names.

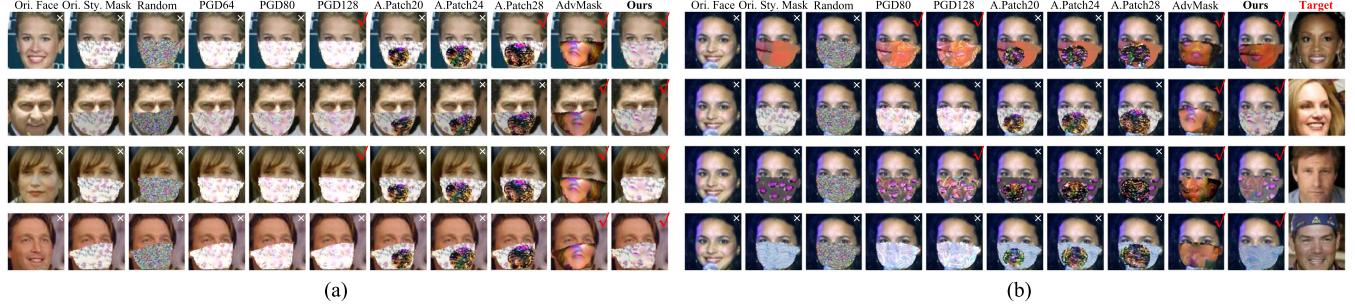


Fig. 4. Adversarial masked face images generated by different attack methods, where images with red check marks on the right top denote successful attacks against FR models, while white crosses denote failed attacks. Note that masks by AdvMask and our method both show strong attack performances, while our masks provide much better stealthiness (much higher SSIM) than AdvMask. (a) Adversarial examples for the target Patricia Hearst (cf. the image in the second row, the right-most column in Fig. 4(b)). (b) Adversarial examples for four targets: Vivica Fox, Patricia Hearst, Aaron Eckhart, Steve Park (from top to bottom).

TABLE IV
AVERAGE ASR AND SSIM OF ATTACK EXPERIMENTS OF 50 TARGETS

# of Targets	Metric	No Mask	Ori Mask	Rand	PGD-64	PGD-80	PGD-128	AdvPatch-20	AdvPatch-24	AdvPatch-28	AdvMask	Ours
5	ASR	4.6%	6.3%	4.2%	39.9%	48.1%	52.4%	7.2%	8.6%	8.1%	88.4%	90.6%
	SSIM	-	-	0.7205	0.8966	0.8831	0.8430	0.8320	0.8031	0.7718	0.6698	0.9177
10	ASR	6.9%	6.0%	5.2%	41.9%	47.7%	47.6%	7.9%	8.5%	9.0%	87.2%	89.8%
	SSIM	-	-	0.7205	0.8951	0.8810	0.8432	0.8345	0.8044	0.7749	0.6702	0.9150
15	ASR	7.8%	4.3%	5.3%	47.9%	53.0%	53.8%	8.1%	8.3%	8.8%	86.5%	90.1%
	SSIM	-	-	0.7203	0.8948	0.8805	0.8431	0.8380	0.8066	0.7748	0.6703	0.9153
20	ASR	7.7%	3.4%	4.4%	46.4%	50.6%	53.0%	6.8%	7.0%	7.4%	85.8%	90.5%
	SSIM	-	-	0.7207	0.8948	0.8808	0.8431	0.8392	0.8078	0.7782	0.6709	0.9155
30	ASR	7.2%	3.3%	3.3%	42.1%	46.2%	49.4%	5.3%	5.4%	5.8%	80.6%	90.8%
	SSIM	-	-	0.7206	0.8948	0.8812	0.8433	0.8409	0.8094	0.7786	0.6668	0.9162
40	ASR	6.6%	3.6%	2.9%	39.6%	43.1%	45.7%	4.7%	4.7%	5.1%	76.2%	90.0%
	SSIM	-	-	0.7203	0.8955	0.8821	0.8443	0.8414	0.8093	0.7781	0.6663	0.9167
50	ASR	6.1%	3.5%	2.4%	36.4%	39.2%	41.6%	4.2%	4.2%	4.6%	69.8%	89.8%
	SSIM	-	-	0.7201	0.8956	0.8822	0.8442	0.8410	0.8094	0.7779	0.6670	0.9170

The best ASR and SSIM results are stressed in bold.

2) Black-Box Attack

Besides, we conducted experiments to evaluate the ASR of our SASMask on FR models that it was not trained on. Specifically, we conducted experiments to examine the attack transferability of different attack methods in the following aspects: model backbone, model head, as well as dataset. Without loss of generality, Vivica Fox was selected herein as the target identity; PGD-128 and AdvPatch-28 were chosen for these experiments.

a) *Backbone transferability*: To evaluate the transferable attack performance from one backbone to another, we conducted backbone transferability experiments of different attack methods. Hereof, we fixed the model head as ArcFace and used ResNets with different depths (34, 50, 101), MobileNet, and GhostNet as our model backbones. Besides, for the comparison

TABLE V
AVERAGE TRANSFER ASR OF ADVERSARIAL MASKED FACE EXAMPLES GENERATED BY DIFFERENT ATTACK METHODS WITH DIFFERENT BACKBONES

Backbone	Rand	PGD	AdvPatch	AdvMask	Ours
ResNet-34 [51]	1.1%	4.1%	2.2%	76.3%	87.5%
ResNet-50 [51]	0.3%	13.1%	1.5%	61.9%	74.1%
ResNet-101 [51]	0.6%	20.0%	1.8%	57.3%	71.5%
MobileNet [52]	0.7%	2.0%	2.0%	41.2%	43.4%
GhostNet [53]	0.2%	5.8%	1.4%	68.0%	72.7%

The significance of bold values are the best results among the experiment results.

methods PGD and AdvPatch, we chose the strongest PGD-128 and AdvPatch-28. Note that we only showed the **average** transfer ASR from one trained backbone to the other four unseen backbones. The experimental results are shown in Table V, from

TABLE VI

AVERAGE TRANSFER ASR OF ADVERSARIAL MASKED FACE EXAMPLES GENERATED BY DIFFERENT ATTACK METHODS WITH DIFFERENT HEADS

Head	Rand	PGD	AdvPatch	AdvMask	Ours
ArcFace [7]	0.3%	2.0%	0.4%	43.2%	51.7%
CosFace [6]	0.0%	18.7%	0.4%	68.2%	74.0%
CircleLoss [54]	0.1%	11.1%	0.8%	70.5%	76.5%
CurricularFace [55]	0.1%	7.6%	0.6%	75.1%	72.4%
MagFace [56]	0.7%	8.5%	0.5%	68.2%	71.2%

The significance of bold values are the best results among the experiment results.

TABLE VII

TRANSFER ASR AND SSIM OF ADVERSARIAL MASKED FACE IMAGES GENERATED BY DIFFERENT ATTACK METHODS FROM ONE DATASET TO THE OTHER THREE DATASETS

Dataset		LFW	VGGFace2	AgeDB	CFP
Rand	ASR	0.3%	3.1%	1.2%	0.5%
	SSIM	0.7166	0.7380	0.7550	0.7910
PGD	ASR	51.1%	82.8%	81.1%	69.4%
	SSIM	0.8545	0.8628	0.8724	0.8917
AdvPatch	ASR	5.1%	10.5%	5.6%	5.4%
	SSIM	0.7590	0.7623	0.7827	0.8080
AdvMask	ASR	98.6%	100.0%	100.0%	99.9%
	SSIM	0.6614	0.6620	0.6792	0.6931
Ours	ASR	98.8%	99.9%	100.0%	99.9%
	SSIM	0.9130	0.9154	0.9197	0.9334

The significance of bold values are the best results among the experiment results.

which we can find that our attack method has the highest attack transferability between seen and unseen models. Besides, we can see that Random masks, PGD masks, and AdvPatch masks have little attack transferability from one model backbone to another. For AdvMask and our method, adversarial examples by small backbones have better attack transferability than those by big models, especially for similar backbones (like ResNets). Such observation is consistent with the view presented in [67].

b) *Head transferability*: Furthermore, transferable experiments between different model heads were performed to examine the transferability of different attack methods. Hereof, we fixed the model backbone as ResNet-50 and utilized one of the following heads to train the models: ArcFace, CosFace, CircleLoss, CurricularFace, and MagFace. As shown in Table VI, our proposed method shows the best transferability between different heads in most cases. Nevertheless, the transferability is not even, e.g., the transferable ASR of ArcFace is much lower than the others, probably because the similarity between ArcFace and the other four heads is lower than that between the other four heads.

c) *Dataset transferability*: We also evaluated the black-box ASR from one dataset to another dataset. Hereof, we trained different attack methods with the ResNet-50 and ArcFace on the LFW dataset and tested it on the VGGFace2, AgeDB, and CFP datasets. The experimental results are displayed in Table VII, where results on the LFW dataset are the white-box attack results, while the others are the transferable attack results. We can see that even though tested on data of different distributions from the trained ones, our SASMask and AdvMask both show

TABLE VIII
COMPARISON WITH FACE ATTACK METHODS ON LFW DATASET

Method	PGDFace	AdvHat	AdvGlass	PatchFace	AdvMakeup	Ours
ASR	14.6%	20.4%	12.4%	10.3%	22.0%	43.4%

The significance of bold values are the best results among the experiment results.

excellent attack transferability, but only our method achieves great stealthiness (high SSIM).

d) *Runtime comparison*: To evaluate the efficiency of every method, we conducted runtime experiments in a 2080Ti GPU to record the inference cost of every method. The results are: PGD costs 4.0712 seconds; AdvPatch costs 0.2999 seconds; AdvMask costs 0.3339 seconds; our SASMask costs 0.4302 seconds. For the results, forward algorithms (AdvPatch, AdvMask, and SASMask) are faster than backward algorithms (PGD) by a great margin. Besides, compared with AdvPatch and AdvMask, our method has an extra adversarial style mask generation process, thus our method is a little slower than the other two forward algorithms. However, our method is still very efficient for inference.

e) *Comparison with other face attack methods*: As mentioned before, we compared our method with five other face attack methods. Here, we used the average transfer ASR of MobileNet. The results are shown in Table VIII. Compared with other attacks, our method achieves the best ASR and outweighs the competitors by a large margin. Note that the PGD in PGDFace acts on the whole facial image, for fair comparison and enough definition of images, we set the ϵ much less than that on the mask region, which is 8/255.

C. ABLATION STUDY

In this subsection, we performed a series of experiments to analyze the following two aspects of our SASMask: 1) style set; 2) style selection; 3) stealthy loss [i.e., \mathcal{L}_1 loss, TV loss, content loss, and style loss]; 4) maximum perturbation size for PGD and AdvPatch.

a) *Style ablation*: In real-world scenarios, masks with single colors, such as white or blue, maybe the most common. We wondered about the effectiveness of the pre-defined style set. Thus, in this ablation experiment, we used four common mask colors (blue, white, black, and red) as the starting optimization points to train adversarial masks. To keep a degree of stealthiness, we set the maximum perturbation as 128/255. The results are displayed in Table XII, which shows that adversarial masks optimized from a single color can maintain good stealthiness, while they cannot achieve effective adversarial attacks. The sub-optimal starting points of single colors may be the main factor that leads to the ineffective adversarial performance of the generated adversarial masks.

b) *Style optimization ablation*: In this ablation experiment, we evaluated the attack strength and the stealthiness of the optimized style generated by our method and the manually selected style. The results are exhibited in Fig. 5, from which we can find that the style optimizer helps us select the optimal adversarial

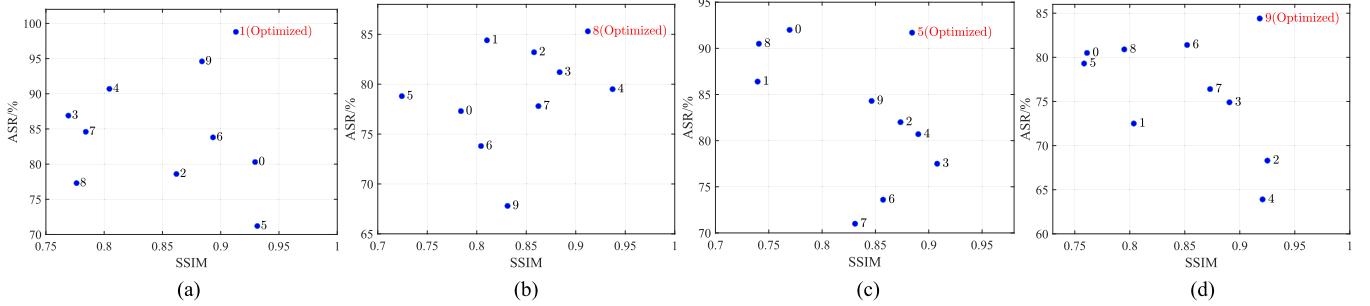


Fig. 5. Trade-off (ASR and SSIM) of adversarial masked face images via manual style selection (style ablation) and style optimization. The point in the upper right of the figure indicates better trade-off. The optimized style selection is highlighted in red bold font. (a) Target: Vivica Fox. (b) Target: Aaron Eckhart. (c) Target: Aaron Eckhart. (d) Target: Steve Park.

TABLE IX
ASR AND SSIM OF ADVERSARIAL MASKED FACE IMAGES BY ABLATING
DIFFERENT STEALTHY LOSS PARTS

\mathcal{L}_{adv}	\mathcal{L}_1	\mathcal{L}_{tv}	\mathcal{L}_c	\mathcal{L}_s	ASR	SSIM
✓	✓	✓	✓	✓	98.8%	0.9130
✓	-	✓	✓	✓	97.6%	0.8337
✓	✓	-	✓	✓	98.3%	0.8437
✓	✓	✓	-	✓	98.6%	0.8837
✓	✓	✓	✓	-	98.3%	0.8993
✓	-	-	✓	✓	98.7%	0.7886
✓	✓	✓	-	-	98.5%	0.8397
✓	-	-	-	-	99.7%*	0.6255

*ASR without any constraint loss is the ASR **upper bound** in our setting, which has an adverse side effect of very low SSIM.

The best ASR and SSIM results are stressed in bold.

style from the style set that achieves a better trade-off between the attack strength and the stealthiness.

c) *Loss ablation*: We further ablated the stealthy loss (i.e., \mathcal{L}_1 , \mathcal{L}_{tv} , \mathcal{L}_c and \mathcal{L}_s) to analyze the attack performance of the SASMask. Table IX illustrates the loss ablation results, from which we can see that ablating any part of the stealthy loss decreases the SSIM by a great margin without little profit of adversarial strength.

d) *Maximum perturbation size*: For PGD and AdvPatch, we selected different maximum perturbation sizes to see the effect of the maximum perturbation size on the attack performance of PGD and AdvPatch. Here, we selected 64/255, 80/255, and 128/255 as the maximum perturbation sizes for PGD; 20, 24, and 28 as the radii of the perturbed adversarial patch circle. The results are shown in Tables III and IV. From the two tables, we can find that the larger perturbation size results in better ASR but lower SSIM.

D. PHYSICAL ATTACK

a) *Offline physical attack*: To verify the effectiveness of SASMask in the physical world, we recruited a group of five female

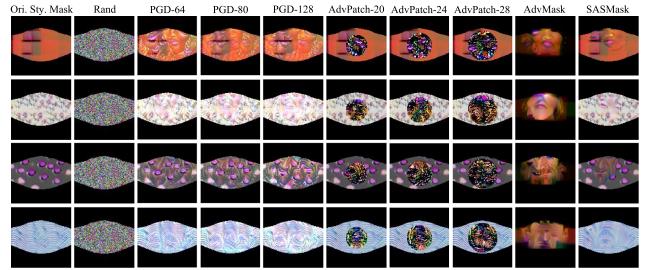


Fig. 6. Adversarial masks generated by different attack methods. From top to bottom, the target identities are Vivica Fox, Patricia Hearst, Aaron Eckhart, and Steve Park, respectively.

TABLE X
ASR WITHOUT OR WITH DIFFERENT TYPES OF MASKS

	Female	Male	All
w/o mask	2.00%	0.00%	1.00%
w/ surgical masks	4.00%	2.00%	3.00%
w/ fashionable masks	6.50%	8.00%	7.25%
w/ AdvMask	77.50%	76.00%	76.75%
w/ SASMask	86.00%	84.50%	85.25%

The significance of bold values are the best results among the experiment results.

and five male participants with the approval of the ethics committee. Firstly, we printed the generated masks on paper and selected six types of masks (one blue surgical mask, four fashionable masks, and the adversarial mask) as our counterparts. For each physical experiment, we shot a short video and chose 10 frames as our test data. Thus, we collected 1400 images (700 female images and 700 male images): 100 without any mask, 100 with the surgical mask, 400 with fashionable masks, 400 with AdvMask, and 400 with SASMask. For a type of masked face, the test data of a participant contains 3 left-side images, 3 right-side images, and 4 front-side images (10 images in total). Then, we used the MTCNNs [68] to detect and align faces. Later, we fed the processed data (some examples are shown in Figs. 6, 7, and 8) into the victim FR model. Here, we used ResNet-50 with ArcFace as the victim FR model. The results are shown in Table X, from which we can see that common masks are hardly

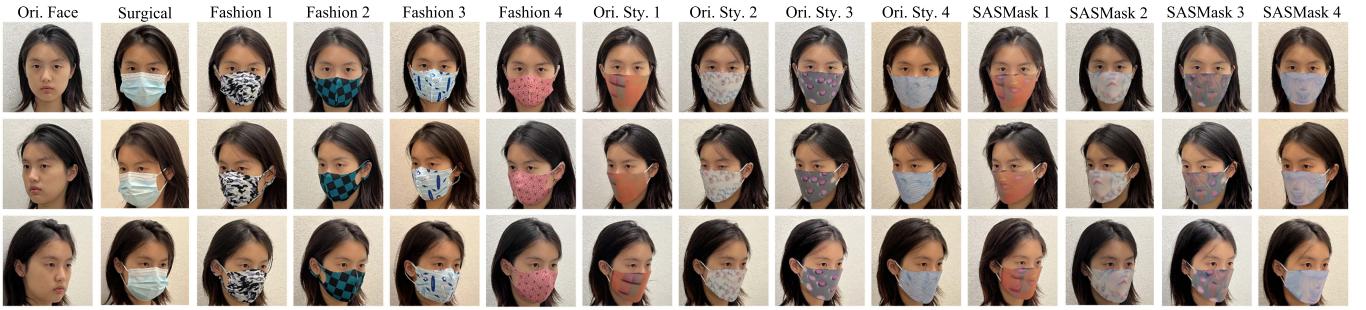


Fig. 7. Physical (adversarial) masked facial examples of one female participant, where photos in each row are taken from different sides: front, left, and right.



Fig. 8. Physical (adversarial) masked facial examples of one male participant, where photos in each row are taken from different sides: front, left, and right.

TABLE XI
SCORES (SCORE DIFFERENCES FROM THE CLEAN FACE IMAGES) OF
ADVERSARIAL FACE IMAGES GENERATED BY DIFFERENT METHODS AGAINST
ONLINE PLATFORMS

	Baidu	iFLYTEK	Tencent
Clean	9.39	49.26	11.01
PGDFace	9.41 (0.02↑)	51.78 (2.52↑)	10.95 (-0.06↓)
PatchFace	20.24 (10.85↑)	65.77 (16.51↑)	21.53 (10.52↑)
AdvGlass	22.17 (12.78↑)	67.58 (18.32↑)	20.95 (9.94↑)
AdvHat	14.64 (5.25↑)	59.85 (10.59↑)	15.19 (4.18↑)
AdvMakeup	28.06 (18.67↑)	77.24 (27.98↑)	27.60 (16.59↑)
AdvMask	31.77 (22.38↑)	87.38 (38.12 ↑)	35.18 (24.17↑)
Ours	33.86 (24.47 ↑)	85.86 (36.60↑)	38.23 (27.22 ↑)

The significance of bold values are the best results among the experiment results.

TABLE XII
ASR AND SSIM OF ADVERSARIAL FACE IMAGES WITH ADVERSARIAL MASKS
OPTIMIZED FROM FOUR SINGLE COLORS AND OUR STYLE SET WITH
STYLE OPTIMIZATION

Start Color	Blue	White	Black	Red	Our Style Set
ASR	0.4%	0.1%	0.1%	0.3%	98.8%
SSIM	0.9432	0.9538	0.9496	0.9551	0.9130

The significance of bold values are the best results among the experiment results.

aggressive, while AdvMask and our SASMask can deceive the FR models, where our method achieves a higher ASR both for females and males.

b) *Online platform attack*: Besides, we used our method and the competitors to generate adversarial face images with the LFW test set and attacked three online commercial FR platforms: Baidu [69], iFLYTEK [70], Tencent [71]. The results are displayed in Table XI, from which we can see that our method

almost achieves the best attack performance among all the listed methods.

V. CONCLUSION

In this paper, we proposed a new stealthy adversarial style mask to attack FR models in both digital and physical settings. Specifically, we trained an adversarial style mask generator that hides adversarial perturbations inside the style masks. Moreover, we proposed to optimize the selection of style to mitigate the sub-optimization of one single style. Compared with existing adversarial works, our method provides both better stealthiness and adversarial strength. Extensive digital experiments, both in white-box settings as well as black-box settings, demonstrated the effectiveness of our proposed SASMask. Furthermore, we printed the SASMask and let participants wear it to attack real-world FR models. In a nutshell, this paper emphasized the vulnerability of deep FR models that they will fail when one wears an elaborated stealthy adversarial style facial mask.

ACKNOWLEDGMENT

The authors acknowledge the use of the National Computational Infrastructure <http://dx.doi.org/10.13039/100010582> (NCI) which is supported by the Australian Government <http://dx.doi.org/10.13039/100015539> and accessed through the Sydney Informatics Hub HPC Allocation Scheme.

REFERENCES

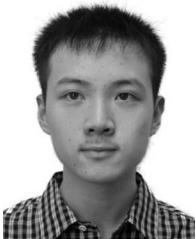
- [1] R. Hadsell, S. Chopra, and Y. LeCun, "Dimensionality reduction by learning an invariant mapping," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, 2006, pp. 1735–1742.

- [2] Y. Sun, Y. Chen, X. Wang, and X. Tang, "Deep learning face representation by joint identification-verification," in *Proc. 28th Int. Conf. Adv. Neural Inf. Proces. Syst.*, 2014, pp. 1988–1996.
- [3] F. Schroff, D. Kalenichenko, and J. Philbin, "FaceNet: A unified embedding for face recognition and clustering," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 815–823.
- [4] Y. Wen, K. Zhang, Z. Li, and Y. Qiao, "A discriminative feature learning approach for deep face recognition," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 499–515.
- [5] W. Liu et al., "Sphereface: Deep hypersphere embedding for face recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 6738–6746.
- [6] H. Wang et al., "CosFace: Large margin cosine loss for deep face recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 5265–5274.
- [7] J. Deng, J. Guo, N. Xue, and S. Zafeiriou, "Arcface: Additive angular margin loss for deep face recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 4690–4699.
- [8] J. Sun, W. Yang, J.-H. Xue, and Q. Liao, "An equalized margin loss for face recognition," *IEEE Trans. Multimedia*, vol. 22, pp. 2833–2843, 2020.
- [9] C. Szegedy et al., "Intriguing properties of neural networks," in *Proc. 2nd Int. Conf. Learn. Representation*, 2014, pp. 1–10.
- [10] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," in *Proc. 2nd Int. Conf. Learn. Representation*, 2014, pp. 1–11.
- [11] S. Amini and S. Ghaemmaghami, "Towards improving robustness of deep neural networks to adversarial perturbations," *IEEE Trans. Multimedia*, vol. 22, pp. 1889–1903, 2020.
- [12] Y. Zhong et al., "Dynamic training data dropout for robust deep face recognition," *IEEE Trans. Multimedia*, vol. 24, pp. 1186–1197, 2022.
- [13] Y. Dong et al., "Efficient decision-based black-box adversarial attacks on face recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 7706–7714.
- [14] Y. Zhong and W. Deng, "Towards transferable adversarial attack against deep face recognition," *IEEE Trans. Inf. Forensics Secur.*, vol. 16, pp. 1452–1466, 2021.
- [15] M. Sharif, S. Bhagavatula, L. Bauer, and M. K. Reiter, "Accessorize to a crime: Real and stealthy attacks on state-of-the-art face recognition," in *Proc. ACM Conf. Comput. Commun. Secur.*, 2016, pp. 1528–1540.
- [16] S. Komkov and A. Petushko, "AdvHat: Real-world adversarial attack on arcface face ID system," in *Proc. 25th Int. Conf. Pattern Recognit.*, 2020, pp. 819–826.
- [17] N. Guetta, A. Shabtai, I. Singh, S. Momiyama, and Y. Elovici, "Dodging attack using carefully crafted natural makeup," 2021. *arXiv:2109.06467*.
- [18] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu, "Towards deep learning models resistant to adversarial attacks," in *Proc. 6th Int. Conf. Learn. Representation*, 2018, pp. 1–10.
- [19] T. B. Brown, D. Mane, A. Roy, M. Abadi, and J. Gilmer, "Adversarial patch," in *NeurIPS 2017 Workshop Mach. Learn. Comput. Secur.*, pp. 1–6, 2017.
- [20] A. Zolfi, S. Avidan, Y. Elovici, and A. Shabtai, "Adversarial mask: Real-world universal adversarial attack on face recognition models," in *Proc. Eur. Conf. Mach. Learn., Knowl. Discov. Databases*, 2022, vol. 3, pp. 304–320.
- [21] R. Duan et al., "Adversarial camouflage: Hiding physical-world attacks with natural styles," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 997–1005.
- [22] Y. Cao, X. Xiao, R. Sun, D. Wang, M. Xue, and S. Wen, "Stylefool: Fooling video classification systems via style transfer," in *Proc. IEEE Symp. Secur. Privacy*, 2023, pp. 1631–1648.
- [23] C. Ding and D. Tao, "Robust face recognition via multimodal deep face representation," *IEEE Trans. Multimedia*, vol. 17, pp. 2049–2058, 2015.
- [24] G. Gao, Y. Yu, H. Lu, J. Yang, and D. Yue, "Context-patch representation learning with adaptive neighbor embedding for robust face image super-resolution," *IEEE Trans. Multimedia*, vol. 25, pp. 1879–1889, 2023.
- [25] S. Chopra, R. Hadsell, and Y. LeCun, "Learning a similarity metric discriminatively, with application to face verification," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, 2005, vol. 1, pp. 539–546.
- [26] S.-M. Moosavi-Dezfooli, A. Fawzi, and P. Frossard, "DeepFool: A simple and accurate method to fool deep neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 2574–2582.
- [27] Y. Cheng et al., "Pasadena: Perceptually aware and stealthy adversarial denoise attack," *IEEE Trans. Multimedia*, vol. 24, pp. 3807–3822, 2021.
- [28] H. Zhang, H. Zhou, N. Miao, and L. Li, "Generating fluent adversarial examples for natural languages," in *Proc. 57th Conf. Assoc. Comput. Linguistics*, pp. 5564–5569, 2019.
- [29] C. Xie et al., "Adversarial examples for semantic segmentation and object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 1378–1387.
- [30] Y. Qin, N. Carlini, G. Cottrell, I. Goodfellow, and C. Raffel, "Imperceptible, robust, and targeted adversarial examples for automatic speech recognition," in *Proc. 36th Int. Conf. Mach. Learn.*, 2019, pp. 5231–5240.
- [31] H. Hosseini and R. Poovendran, "Semantic adversarial examples," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops*, 2018, pp. 1614–1619.
- [32] C. Laidlaw and S. Feizi, "Functional adversarial attacks," in *Proc. 33rd Adv. Neural Inf. Proces. Syst.*, 2019, pp. 10408–10418.
- [33] X. Zeng et al., "Adversarial attacks beyond the image space," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 4297–4306.
- [34] H.-T. D. Liu, M. Tao, C.-L. Li, D. Nowrouzezahrai, and A. Jacobson, "Beyond pixel norm-balls: Parametric adversaries using an analytically differentiable renderer," in *Proc. 7th Int. Conf. Learn. Representation*, 2019, pp. 1–12.
- [35] Y.-C.-T. Hu et al., "Naturalistic physical adversarial patch for object detectors," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 7848–7857.
- [36] L. Gao, Z. Huang, J. Song, Y. Yang, and H. T. Shen, "Push & pull: Transferable adversarial examples with attentive attack," *IEEE Trans. Multimedia*, vol. 24, pp. 2329–2338, 2022.
- [37] C. Wan, F. Huang, and X. Zhao, "Average gradient-based adversarial attack," *IEEE Trans. Multimedia*, pp. 1–14, 2023.
- [38] A. Kurakin, I. Goodfellow, and S. Bengio, "Adversarial examples in the physical world," in *Proc. 4th Int. Conf. Learn. Representation*, 2016, pp. 1–15.
- [39] A. Athalye, L. Engstrom, A. Ilyas, and K. Kwok, "Synthesizing robust adversarial examples," in *Proc. 35th Int. Conf. Mach. Learn.*, 2018, pp. 284–293.
- [40] K. Eykholt et al., "Robust physical-world attacks on deep learning visual classification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 1625–1634.
- [41] R. Duan et al., "Adversarial laser beam: Effective physical-world attack to DNNs in a blink," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 16062–16071.
- [42] Z. Hu et al., "Adversarial texture for fooling person detectors in the physical world," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 13307–13316.
- [43] Y. Zhong, X. Liu, D. Zhai, J. Jiang, and X. Ji, "Shadows can be dangerous: Stealthy and effective physical-world adversarial attack by natural phenomenon," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 15345–15354.
- [44] Z. Xiao et al., "Improving transferability of adversarial patches on face recognition with generative models," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 11845–11854.
- [45] B. Yin et al., "Adv-makeup: A new imperceptible and transferable attack on face recognition," in *Proc. Int. Joint Conf. Artif. Intell.*, 2021, pp. 1252–1258.
- [46] L. A. Gatys, A. S. Ecker, and M. Bethge, "Image style transfer using convolutional neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 2414–2423.
- [47] J. Johnson, A. Alahi, and L. Fei-Fei, "Perceptual losses for real-time style transfer and super-resolution," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 694–711.
- [48] V. Dumoulin, J. Shlens, and M. Kudlur, "A learned representation for artistic style," in *Proc. 5th Int. Conf. Learn. Representation*, 2017, pp. 1–11.
- [49] Y. Feng, F. Wu, X. Shao, Y. Wang, and X. Zhou, "Joint 3D face reconstruction and dense alignment with position map regression network," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 534–551.
- [50] D. P. Kingma and J. L. Ba, "Adam: A method for stochastic optimization," in *Proc. 3rd Int. Conf. Learn. Representation*, 2015, pp. 1–11.
- [51] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 770–778.
- [52] S. Chen, Y. Liu, X. Gao, and Z. Han, "MobileFaceNets: Efficient CNNs for accurate real-time face verification on mobile devices," in *Proc. 13th Chin. Conf. Biomed. Recognit.*, 2018, pp. 428–438.

- [53] K. Han et al., "GhostNet: More features from cheap operations," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 1580–1589.
- [54] Y. Sun et al., "Circle loss: A unified perspective of pair similarity optimization," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 6398–6407.
- [55] Y. Huang et al., "CurricularFace: Adaptive curriculum learning loss for deep face recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 5901–5910.
- [56] Q. Meng, S. Zhao, Z. Huang, and F. Zhou, "MagFace: A universal representation for face recognition and quality assessment," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 14225–14234.
- [57] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Proc. 18th Int. Conf. Med. Image Comput. Comput. Assist. Interv.*, 2015, pp. 234–241.
- [58] G. B. Huang, M. Mattar, T. Berg, and E. Learned-Miller, "Labeled faces in the wild: A database for studying face recognition in unconstrained environments," University of Massachusetts, Amherst, Tech. Rep., pp. 1–11, 2007.
- [59] Q. Cao, L. Shen, W. Xie, O. M. Parkhi, and A. Zisserman, "VGGFace2: A dataset for recognising faces across pose and age," in *Proc. IEEE 13th Int. Conf. Autom. Face Gesture Recognit.*, 2018, pp. 67–74.
- [60] S. Moschoglou et al., "AgeDB: The first manually collected, in-the-wild age database," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops*, 2017, pp. 1997–2005.
- [61] S. Sengupta et al., "Frontal to profile face verification in the wild," in *Proc. IEEE Winter Conf. Appl. Comput. Vis.*, 2016, pp. 1–9.
- [62] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, Apr. 2004.
- [63] T.-Y. Lin et al., "Microsoft COCO: Common objects in context," in *Proc. Eur. Conf. Comput. Vis.*, 2014, vol. 8693, pp. 740–755.
- [64] D. Yi, Z. Lei, S. Liao, and S. Z. Li, "Learning face representation from scratch," 2014, *arXiv:1411.7923*.
- [65] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proc. 3rd Int. Conf. Learn. Representation*, 2015, pp. 1–10.
- [66] L. Rice, E. Wong, and Z. Kolter, "Overfitting in adversarially robust deep learning," in *Proc. 37th Int. Conf. Mach. Learn.*, 2020, pp. 8093–8104.
- [67] L. Wu, Z. Zhu, C. Tai, and E. Weinan, "Understanding and enhancing the transferability of adversarial examples," Peking Univ., Beijing, China, Tech. Rep., pp. 1–13, 2018.
- [68] K. Zhang, Z. Zhang, Z. Li, and Y. Qiao, "Joint face detection and alignment using multitask cascaded convolutional networks," *IEEE Signal Process. Lett.*, vol. 23, no. 10, pp. 1499–1503, Oct. 2016.
- [69] Baidu, "Online face verification platform," 2023. [Online]. Available: <https://cloud.baidu.com/doc/FACE/index.html>
- [70] iFLYTEK, "Online face verification platform," 2023. [Online]. Available: <https://global.xfyun.cn/>
- [71] Tencent, "Online face verification platform," 2023. [Online]. Available: <https://cloud.tencent.com/product/facerecognition>



Huihui Gong is currently working toward the Ph.D. degree with the School of Computer Science, Faculty of Engineering, The University of Sydney, Sydney, NSW, Australia. His research interests include adversarial machine learning and vulnerability detection on software.



Minjing Dong received the M.Phil. degree from The University of Sydney, Camperdown, NSW, Australia, where he is currently working toward the Ph.D. degree with the School of Computer Science. His research interests include human behavior analysis, adversarial robustness, and neural architecture search.



Siqi Ma received the B.S. degree in computer science from Xidian University, Xi'an, China, in 2013, and Ph.D. degree in information system from Singapore Management University, Singapore, in 2018. She was a Research Fellow of a distinguished system security group from CSIRO and was a Lecturer with The University of Queensland, Brisbane, QLD, Australia. She is currently a Senior Lecturer with the University of New South Wales, Canberra Campus, Kensington, NSW, Australia. Her research interests include data security, IoT security, and software security.



Seyit Camtepe (Senior Member, IEEE) received the Ph.D. degree from Rensselaer Polytechnic Institute, Troy, NY, USA, in 2007. He is currently a Principal Research Scientist and the Team Leader with CSIRO Data61. He was with Technische Universität, Berlin as a Senior Researcher, and QUT as a Lecturer. He was among the first to investigate the security of Android smartphones and inform society of the rising malware threat. His research interests include autonomous security, malware detection and prevention, smartphone security, applied and malicious cryptography, and CII security.



Surya Nepal (Senior Member, IEEE) is currently a Senior Principal Research Scientist with CSIRO's Data61. He leads the distributed systems security group comprising more than 30 research staff and more than 50 postgraduate students. He is also the Team Leader of the Cybersecurity Cooperative Research Centre, a national initiative in Australia. He has more than 300 peer-reviewed publications to his credit. His main research interests include the development and implementation of technologies in the area of cybersecurity and privacy, and AI and cybersecurity. He is a Member of the Editorial Board of *IEEE TRANSACTIONS ON SERVICES COMPUTING*, *ACM Transactions on Internet Technology*, *IEEE TRANSACTIONS ON DEPENDABLE AND SECURE COMPUTING* and *Frontiers of Big Data Security Privacy and Trust*.



Chang Xu (Senior Member, IEEE) is currently a Senior Lecturer with the School of Computer Science, The University of Sydney, Camperdown, NSW, USA. He has authored or coauthored more than 100 papers in prestigious journals and top-tier conferences. His research interests include machine learning algorithms and related applications in computer vision. He was the recipient of several paper awards, including the Distinguished Paper Award in AAAI 2023, Best Student Paper Award in ICDM 2022, Best Paper Candidate in CVPR 2021, and Distinguished Paper Award in IJCAI 2018. He was the Area Chair of NeurIPS, ICML, ICLR, KDD, CVPR, and MM, and a Senior PC Member of AAAI and IJCAI. In addition, he was an Associate Editor with *IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE*, *IEEE TRANSACTIONS ON MULTIMEDIA*, and *Transactions on Machine Learning Research*. He has been named a Top Ten Distinguished Senior PC Member in IJCAI 2017 and an Outstanding Associate Editor at IEEE T-MM in 2022. He was the recipient of The University of Sydney Vice-Chancellor's Award for Outstanding Early Career Research.