General Adversarial Perturbation Simulating: Protect Unknown System by Detecting Unknown Adversarial Faces

Hefei Ling*
Huazhong University of Science and Technology
Wuhan, China
lhefei@hust.edu.cn

Feiran Sun
Huazhong University of Science and Technology
Wuhan, China
frsun@hust.edu.cn

Jinyuan Zhang
Industrial and Commercial Bank of China
Guangzhou, China
zhangjy@sdc.icbc.com.cn

Xiaorui Lin
Industrial and Commercial Bank of China
Guangzhou, China
linxr@sdc.icbc.com.cn

Jiazhong Chen
Huazhong University of Science and Technology
Wuhan, China
jzchen@hust.edu.cn

Ping Li
Huazhong University of Science and Technology
Wuhan, China
lpshome@hust.edu.cn

Qian Wang
Huazhong University of Science and Technology
Wuhan, China
yqwq1996@hust.edu.cn

2023 International Joint Conference on Neural Networks (IJCNN) | 978-1-6654-8867-9/23/$31.00 ©2023 IEEE | DOI: 10.1109/IJCNN54540.2023.10191564

***Abstract*—Benefitting from the development of convolutional neural networks (CNNs), face recognition systems (FRSs) play a key role in many security-critical systems. However, FRSs have been proved to be vulnerable to adversarial faces (adv-faces). Adv-faces aim to change classification results by adding a subtle perturbation on real faces. The existence of adv-faces poses a significant threat to financial and privacy security. Previous detection methods require either training on pre-computed adv-faces or accessing to protected victim FRSs, bringing a dilemma in practical using. In this work, we heuristically propose an adversarial face detection method called General Adversarial Perturbation Simulating (GAPS) which is blind to both adversarial attacks and FRSs. Simulating noise patterns of several gradient-based adversarial perturbations, GAPS is able to generate simulated adversarial faces (sadv-faces) guiding detectors to learn general adversarial perturbation features and focus on classifying sensitive regions. Extensive experiments on LFW and CASIA-WebFace show that our method outperforms 9 state-of-the-art baseline methods and demonstrate the effectiveness of GAPS.**

***Index Terms*—Adversarial face detection, Adversarial example defense, Face recognition security**

I. Introduction

Convolutional neural networks (CNNs) have achieved impressive performance on a wide range of tasks including image classification [1], person re-identification [2], [3], [4] and face recognition [5], [6], [7]. However, recent studies [8], [9] have shown that CNNs are vulnerable to adversarial examples, in which the adversarial perturbations added to adversarial examples are generally imperceptible to the sense of humanity, yet can cause severe output errors. As a branch of adversarial examples, adversarial faces (adv-faces) pose a risk to the use of face recognition technology and threats property security and privacy security. As a kind of the defense strategies, Adversarial face detection methods [10], [11], [12] are proposed to defend against adv-faces by training adversarial face detectors. These methods are usually designed for specific attacks or specific tasks resulting in unsatisfied detection generalization, and most of them require to modify or access the protected FRSs which is inapplicable to practical use.

In this paper, we proposed General Adversarial Perturbation Simulating (GAPS) consisting of Random Adversarial Patch (RAP) and Representative Forgery Mining (RFM) to detect adv-faces.We analyzed adversarial perturbations generated by several gradient-based adversarial attacks and discovered the fixed patterns of adversarial perturbations. These fixed patterns enable us to produce a vast of simulated adv-faces (sadv-faces) forming a closure of adv-faces in feature space. According to the above analysis, we heuristically proposed Random Adversarial Patch (RAP) which simulates a general noise pattern and generates sadv-faces. Trained on sadv-faces and real face images, detectors are able to learn a general representation for adv-faces which generated from unseen attacks. In order to improve detection performance, we incorporated Representative Forgery Mining (RFM) [13] into our method. RFM is a data augmentation method based on attention mechanism in the field of Face Forgery Detection, guiding detectors to focus on classifying sensitive regions. RFM consists of two components: forgery attention map (FAM) and suspicious forgeries erasing (SFE).

GAPS is able to detect adversarial attacks with neither identifying attacks type nor accessing to the protected FRS, and can be easily integrated with various CNNs without extra structure modification and dataset reconstruction. The differences between the previous method and our method are shown in Figure 1.

*Corresponding author: Hefei Ling (lhefei@hust.edu.cn)

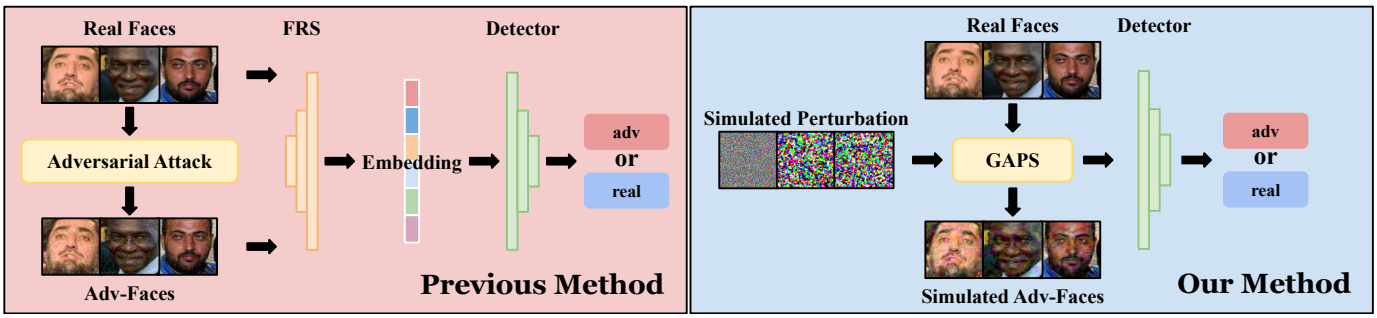


Fig. 1. Previous Adversarial face detection methods require to access the protected FRS or need to know specific attack method for training, bringing a dilemma in practical use. Training on simulated adversarial faces rather than adv-face from specific attack, our method GAPS is able to defend against adversarial attacks with neither identifying attacks type nor accessing to the protected FRS.

Extensive experiments on LFW [14] and CASIA-WebFace [15] show that our method outperforms 9 state-of-the-art baseline [12], [16], [17], [18], [19], [20], [21], [22], [23] methods and demonstrate that the effectiveness of GAPS.

The main contributions of this work are as follows:

- Analyzing adversarial perturbations, we discovered the fixed patterns of adversarial perturbations and heuristically proposed General Adversarial Perturbation Simulating (GAPS) composed by Random Adversarial Patch (RAP) and Representative Forgery Mining (RFM) to detect adversarial faces (adv-faces) where RAP generates simulated adv-faces and RFM guide detector to focus on classifying sensitive regions.
- Trained only on real faces and simulated adversarial faces (sadv-faces) generated by RAP, our method neither need pre-computed adv-faces nor require to modify or access protected facial recognition systems (FRSs), satisfied the demand of practical using.
- Numerous of experiments on LFW and CASIA-WebFace datasets demonstrated the satisfactory adv-faces detection performance of our method.

II. Related Work

A. Adversarial Attack

Adversarial attack is to craftily manipulate normal examples with imperceptible perturbation to fool classifiers and make classifiers misclassify.

Goodfellow et al. [9] proposed FGSM to search for a feasible solution along the negative gradient sign direction of the cost function. Kurakin et al. [24] proposed BIM by adopting an iterative searching strategy to improve the attack performance. Madry et al. [25] suggested a powerful attack PGD, approximating the optimal solution of saddle point (min-max) formulation. To improve the transferability of adversarial examples, Dong et al. [26] proposed MI-FGSM by integrating a momentum term into the iterative attack method. Concentrating on producing transferable and imperceptible forgery faces , Yang et al. [27] proposed TIP-IM to generate adversarial faces against black-box face recognition systems, bringing new challenge for Adversarial face detection. In this work, our method is proposed to detect adv-faces generated by all above attack methods.

B. Adversarial face detection

Generally, adversarial face defense strategies can be roughly divided into three categories: data preprocessing, adversarial training and adversarial face detection. Data preprocessing such as randomization and compression [28], [29] performs operations on input images to weaken adversarial perturbations but has been proven to be insufficient to defend against adversarial attacks. Adversarial training is achieved by modifying model architecture, training method [30] and regularization [31]. Most of these methods require to use pre-computed adv-faces and retrain model after novel attacks appear.

Adversarial face detection methods aims to detect and filter out adv-faces to protect FRS. Tao et al. [10] proposed Attacks meet Interpretability (AmI) that utilizes interpretability of FRS to detect adv-faces. Deng et al. [11] proposed Lightweight Bayesian Refinement (LiBRe) based on leveraging Bayesian neural networks for detection. Massoli et al. [12] proposed a detection technique using Multi-Layer Perceptron and Long-Short Term Memory network and using k-Nearest Neighbor to generate deep features attacks and to guide adversarial face detection. Previous detection methods require training with pre-computed adv-faces or accessing to the protected FRS, which poses a dilemma for practical use. As the same time, these methods are usually designed for specific attacks or specific tasks, resulting in low generalizability on unseen attacks. To solve this problem, we proposed a FRS-agnostic and attack-agnostic method which is trained only on real faces and simulated adv-faces in this work.

C. Fake Face Detection and Data Augmentation

Forgery region localization is one of the efficient techniques for fake face detection. Dang et al. [32] proposed an attention-based detector to locate forgery region and improve detection performance. Wang et al. [13] used forgery attention map and suspicious forgeries erasing to locate forgery classifying sensitive regions.

Data augmentation is a useful approach that addresses underfitting problem caused by insufficient data and prevents

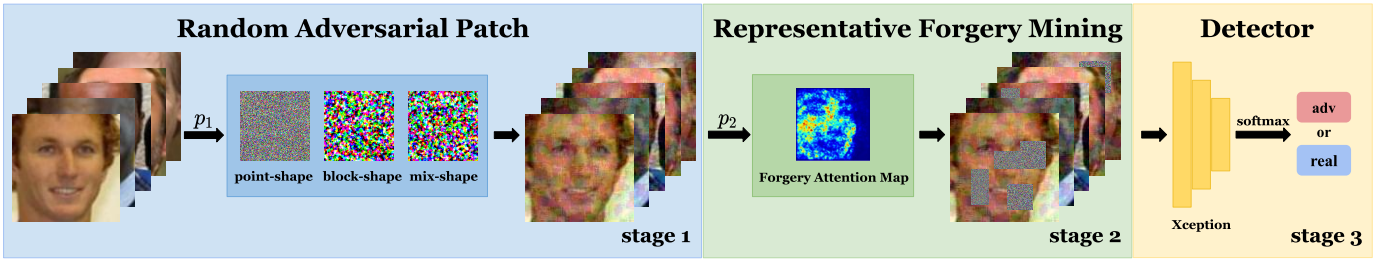


Fig. 2. The pipeline of GAPS can be divided into three steps. Firstly, we use RAP to simulate adversarial perturbations and generate three types of simulated perturbations. Simulated perturbations are added to real faces with probability p_1 to generate sadv-faces. Then, we use RFM [13] with probability p_2 to compute classifying sensitive regions and perform mask operation on these regions. Finally, we use processed images as input to train detectors.

overfitting problem [1], [33]. Random cropping [1] are general data augmentation methods. Applying a mask on image is also a common data augmentation method. Random Erasing [33] uses a random mask to cover randomly selected rectangle regions in image.

In this work, we incorporated Representative Forgery Mining into our method to train detectors focusing on classifying sensitive regions.

III. Proposed Method

We propose a FRS-agnostic and attack-agnostic method General Adversarial Perturbation Simulating (GAPS), which is able to detect adversarial attacks with neither knowing attacks type nor accessing to the protected FRS. GAPS simulates adversarial perturbations to generate simulated adv-faces (sadv-faces) while guiding detectors to focus on classifying sensitive regions to improve detection performance.

As shown in Figure 2, our method is composed of two components: Random Adversarial Patch (RAP) and Representative Forgery Mining (RFM). In this section, we analyzed the fixed patterns of adversarial perturbations and details the process of RAP and RFM. As in the common setting, we treat face adversarial detection as a binary classification problem.

A. Adversarial Attack Perturbations Pattern Analysis

Digital adversarial attacks is adding a minimal perturbation η to the real face image $\mathrm{x}_{\mathrm{real}}$, such that the FRS predicts an incorrect output for the adv-face image $\mathrm{x}_{\mathrm{adv}}$. Generally, the process of generating adv-faces is shown in Formulas 1 and 2, where ϵ controls the magnitude of perturbation, $\mathrm{sign}(\cdot)$ represents the sign function, $\mathcal{L}(\cdot)$ represents the loss function and θ represents the parameters of the model used to generate adv-faces.

$$\mathrm{x}_{\mathrm{adv}} = \mathrm{x}_{\mathrm{real}} + \eta \tag{1}$$

$$\eta = \epsilon \mathrm{sign}\left(\nabla_{\mathrm{x}_{\mathrm{real}}} \mathcal{L}\left(\theta, \mathrm{x}_{\mathrm{real}}, \mathrm{y}_{\mathrm{target}}\right)\right) \tag{2}$$

We use Torchattacks [34] library to generate adv-faces for seven adversarial attack methods on CASIA-WebFace [15] dataset. All adv-faces are generated at following hyper-parameters: maximum perturbation ϵ is set to 5, step size is set to 1, number of iterations for iterative attack method is set to 10, and the rest of parameters are default. We randomly select 1000 images from these adv-faces, analyze and visualize adversarial perturbations respectively, as shown in Figure 3.

We can draw the following conclusions from the figure: 1). Adversarial perturbations added to real images are highly similar and can be view as point shape noise or block shape noise. 2). Noise pattern of point shape and block shape are neither similar to salt-and-pepper noise nor Gaussian noise. This inspired us to speculate that these point and block noise patterns are able to simulate general adversarial noise patterns and plentiful enough random noises may cover noises generated by any attacks.

B. Random Adversarial Patch

Through the above analysis, we propose Random Adversarial Patch (RAP). RAP simulates perturbations of several adversarial attack methods to generate sadv-faces without knowing attacks type.

RAP traverses each pixel of real face image separately and uses stochastic gradient to produce three type of simulated adversarial perturbations: point shape, block shape and mix shape which synthesizes the point shape and block shape:

$$\mathrm{M}_{\mathrm{point,block,mix}} = \epsilon \mathrm{sign}\left(\nabla_{\mathrm{random}}\right) \tag{3}$$

where $\mathrm{M}_{\mathrm{point,block,mix}}$ represents simulated adversarial perturbations generated by RAP, ∇_{random} represents stochastic gradient. Three types of simulated adversarial perturbations generated by RAP are shown in Figure 4

Concretely, as shown in Algorithm 1, the point shape simulated adversarial perturbation adds stochastic gradient to each randomly selected pixel in a probability, while the block shape manipulate on selected squares which are also stochastic. After that RAP restrict perturbations under a normalization of ϵ by clipping operation:

$$\eta_s = \mathrm{Clip}_{[-\epsilon,\epsilon]}(\mathrm{M}_{\mathrm{point,block,mix}}) \tag{4}$$

Attaching simulated adversarial perturbation to real face image, RAP generates a plenty of sadv-faces:

$$\mathrm{x}_{\mathrm{adv}} = \mathrm{x}_{\mathrm{real}} + \eta_s \tag{5}$$

Taking place of adv-faces, sadv-faces act as negative samples and participate in detector training together with real faces. In this way, detector learns a general representation of adversarial noise patterns through sadv-faces without neither

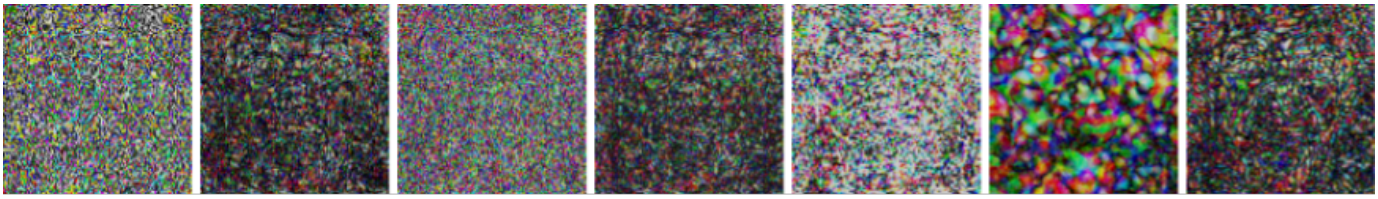


Fig. 3. Average adversarial perturbation over 1000 adversarial faces for seven adversarial attack method, the pixel value of each image is enlarged 20 times.

Algorithm 1 Random Adversarial Patch

Require: Real face image x_{real}; step size α; magnitude of perturbation ϵ; noise type T; image size H, W, C; maximum of the block perturbation side length sl

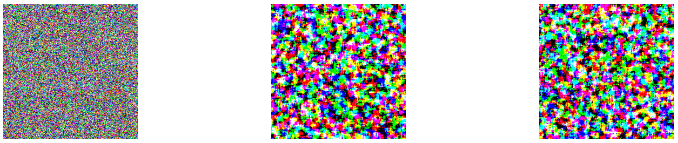
Ensure: Simulated adv-face x_{adv};

Fill M_{point} and M_{block} with 0, M_{point} and M_{block} size is $[H, W, C]$;
for $h = 0$; $h < H$; $h++$ **do**
  for $w = 0$; $w < W$; $w++$ **do**
    Randomly selected gradient rs
    $top = \max(h - sl, 0)$, $bot = \min(h + sl, H)$;
    $lef = \max(w - sl, 0)$, $rig = \min(w + sl, W)$;
    Fill $M_{point}[h, w]$ with $\alpha \times rs$;
    Fill $M_{block}[top : bot, lef : rig]$ with $\alpha \times rs$;
  end for
end for
Clip M_{point}, M_{block} and M_{mix} to $[-\epsilon, \epsilon]$
if T is 'mix' **then**
  $x_{adv} = x_{real} + M_{point} + M_{block}$;
else
  $x_{adv} = x_{real} + M_T$;
end if
Clip x_{adv} to $[0, 255]$;
return x_{adv};

requiring pre-computed adv-faces nor knowing origin of adversarial attacks.

C. Representative Forgery Mining

In order to further improve detection performance, we incorporates Representative Forgery Mining (RFM) [13] into our method. RFM is a data augmentation method based on attention mechanism in the field of Face Forgery Detection, including two components. 1) Forgery Attention Map is the foundation of RFM, which reveals the sensitivity of detectors on each facial region. 2) Based on Forgery Attention Map, Suspicious Forgeries Erasing is applied to augment original images for training detectors. RFM guides detectors to learn adversarial perturbation features more deeply and focus on classifying sensitive regions. During training, each iteration with RFM only needs to propagate forward and backward twice.



(a) point shape (b) block shape (c) mix shape

Fig. 4. Three kinds of simulated adversarial perturbations generated by RAP, the pixel value of each image is enlarged 20 times.

IV. Experiments

In this section, we use two datasets [14], [15] and eight adversarial attack methods [9], [24], [25], [26], [27], [35], [36], [37] to generate adv-faces and demonstrate the effectiveness of GAPS. Experimental results are compared with nine existing methods [12], [16], [17], [18], [19], [20], [21], [22], [23].

A. Dataset

We evaluate our GAPS by performing experiments on LFW [14] and CASIA-WebFace [15] datasets. **LFW** contains 13,233 face images of 5,749 subjects. **CASIA-WebFace** comprises of 494,414 face images from 10,5754 different subjects. In order to facilitate the comparison of face similarity, we consider subjects with at least two face images. After filtering, 9,164 face images of 1,680 subjects in LFW are available for evaluation.

B. Experiment Settings

For CASIA-WebFace [15], we generate adv-faces using pre-trained ArcFace [5] and seven attack methods: FGSM [9], BIM [24], PGD [25], RFGSM [35], MIFGSM [26], TIFGSM [36]

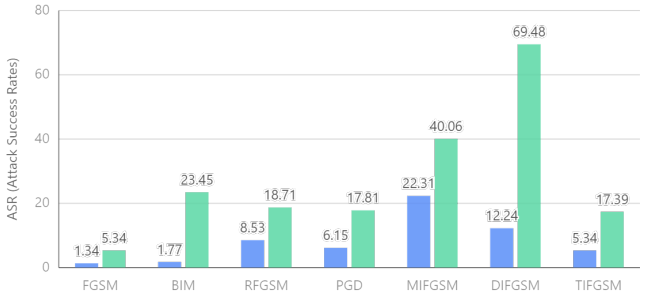


Fig. 5. Attack success rates (ASR) of the adv-faces generated by seven attacks. For each attack, blue represents ASR of adv-faces with $\epsilon = 5$, and green represents ASR of adv-faces with $\epsilon = 10$.

TABLE I
COMPARISON WITH 9 SOTA BASELINE METHODS. OUR GAPS DETECTOR IS TRAINED WITHOUT ANY REAL ADV-FACES, NUMBERS IN THE TABLE REPRESENT ADVERSARIAL DETECTION ACCURACY.

Detector		Year	FGSM [9] Detection ACC	PGD [25] Detection ACC	Mean
General	Gong et al. [16]	2017	98.94	97.91	98.43
	ODIN [17]	2018	83.12	84.39	83.76
	Steganalysis [18]	2019	88.76	89.34	89.05
Face	UAP-D [19]	2018	61.32	74.33	67.83
	SmartBox [20]	2018	58.79	62.53	60.66
	Goswami et al. [21]	2019	84.56	91.32	87.94
	Massoli et al. [12] (MLP)	2020	63.58	76.28	69.93
	Massoli et al. [12] (LSTM)	2020	71.53	76.43	73.98
	Agarwal et al. [22]	2020	94.44	95.38	94.91
	FaceGuard [23]	2021	99.85	99.85	99.85
Ours: GAPS		2022	**99.98**	**99.96**	**99.97**

and DIFGSM [37]. All of adversarial attacks are set as non-target attack, face recognition model and attack parameters for adv-faces generating are same as section III-A. For the selection of maximum perturbation ϵ values, after our pre-experiments, we found that using adv-faces with a large magnitude are too easily to be detected, and which with a small magnitude are unnecessary to take into account due to its extremely low attack accuracy. Thus, we selected $\epsilon = [5, 10]$ for producing adv-faces and sadv-faces. Attack success rates of the adv-faces we used in experiments are shown in the Figure 5. The backbone of detectors is Xception [38]. During model training, we apply random and center cropping and we flip each image horizontally with the probability of 0.5, all images are cropped to 112×112. All detectors are trained with the same number of steps and using Adam [39] optimizer with learning rate of 0.0002 and batch-size is set to 64. We utilize cross entropy loss function with loss term to stabilize training. During model testing, we evaluate the performance of detectors using Detection Accuracy (ACC).

C. Comparison with SOTA face adversarial detection methods

We choose 9 SOTA adversarial detectors proposed for general adversarial detection [16], [17], [18] and adversarial face detection [12], [19], [20], [21], [22], [23], and we conduct experiments on two datasets [14], [15] to demonstrate the effectiveness of GAPS.

All detectors are trained on real faces and adv-faces generated by two adversarial attacks [9], [25] using CASIA-WebFace [15]. Unlike most baselines, our GAPS detector does not utilize any pre-computed adv-faces for training. Subsequently, we use the filtered LFW [14] (9,164 face images of 1,680 subjects) to generate adv-faces for evaluation.

As shown in Table I, we find that compared to all baselines, our GAPS effectively improves detection performance and achieves the highest detection accuracy. Compared with general adversarial detection methods [16], [17], [18], our GAPS is more suitable for adversarial face detection. Detection ACC of our GAPS is 1% to 10% higher than methods above. Moreover, our GAPS achieves superior detection accuracy compared to hand-crafted feature methods [19], [20], at the same time, compared with these methods, our GAPS incorporates feature extraction into model training, which not only simplifies feature extraction but also improves adversarial detection performance. Some methods [12], [21] require access to protected face recognition systems (FRSs) to obtain face feature for adversarial detection, which is not suitable for most systems that require adversarial attack defence. Since most of the above systems are face security critical, we usually do not have permission to access FRSs in this case.

Besides, compared with Massoli's method [12] and FaceGuard [23], our GAPS is more easy to use and integrate. Massoli's method and FaceGuard requires at least two models which are redundant in structure and difficult to train, whereas our GAPS only uses an adversarial detector with simple model structure.

In summary, adversarial face detection performance of our GAPS is superior to all baselines [12], [16], [17], [18], [19], [20], [21], [22], [23]. Our GAPS can defend adversarial attacks without accessing protected FRSs, at the same time, GAPS is easy to use and integrate without complex model structure or building adv-face datasets for training.

V. ABLATION STUDY

We conduct ablation study to verify the performance of GAPS. We use **Adv** to represent training detectors with real faces and adversarial faces (adv-faces) generated by FGSM, use **Real** to represent training detectors with real faces and simulated adv-faces (sadv-faces) generated by RAP. **RAP**, **RFM** and **GAPS** represent using of RAP, RFM and GAPS according to the probability of 0.5 respectively. Combining above settings, we have six data combinations for detector training:

- **Adv**: Training detectors using adv-faces generated by FGSM and real faces.
- **Adv + RAP**: Training detectors using adv-faces generated by FGSM, sadv-faces generated by RAP and real faces. Real faces are augmented with RAP according to the probability of 50% to generate sadv-faces.

TABLE II
ABLATION EXPERIMENT RESULTS, WE USE SIX DATA COMBINATIONS FOR EXPERIMENTS, THE NUMBERS IN THE TABLE REPRESENT DETECTORS DETECTION ACC.

Detector	FGSM [9]	BIM [24]	PGD [25]	RFGSM [35]	MIFGSM [26]	TIFGSM [36]	DIFGSM [37]
Adv	99.9479	95.7919	99.4854	97.4640	99.8196	51.2409	86.9732
Adv + RFM	99.9372	94.7490	99.8102	97.3160	99.9175	50.8236	81.9880
Adv + RAP	99.8757	99.1883	99.8313	99.5813	99.8146	84.9791	98.0527
Adv + GAPS	99.8540	99.8106	99.8513	99.8443	99.8470	85.4856	99.4266
Real + RAP	99.9642	**99.9514**	99.9556	**99.9579**	99.9292	**98.1967**	98.7200
Real + GAPS	**99.9823**	99.5010	**99.9579**	99.7601	**99.9586**	88.5566	**99.8760**

- **Adv + RFM**: Training detectors using adv-faces generated by FGSM and real faces. All faces are augmented with RFM according to the probability of 50%.
- **Adv + GAPS**: Training detectors using adv-faces generated by FGSM, simulated adv-faces (sadv-faces) generated by RAP and real faces. Real faces are augmented with RAP according to the probability of 50% to generate sadv-faces. All faces are augmented with RFM according to the probability of 50%.
- **Real + RAP**: Training detectors only using simulated adv-faces (sadv-faces) generated by RAP and real faces. Real faces are augmented with RAP according to the probability of 50% to generate sadv-faces.
- **Real + GAPS**: Training detectors only using simulated adv-faces (sadv-faces) generated by RAP and real faces. Real faces are augmented with RAP according to the probability of 50% to generate sadv-faces. All faces are augmented with RFM according to the probability of 50%.

A. Effect of Random Adversarial Patch

As shown in Table II, 'Real+RAP' detector trained using sadv-faces generated by RAP and real faces achieves the best performance among seven adversarial attacks. At the same time, by comparing experiment 'Adv' and 'Adv+RAP', the RAP-guided detector have higher performance than the detector without RAP. From the above experiments, we can draw a conclusion that RAP can effectively simulate adversarial perturbations, guide detectors to learn more general adversarial perturbation features and improve the generalizability of detectors.

B. Effect of Representative Forgery Mining

As shown in Table II, we found that using RFM without RAP cannot effectively improve detection performance. After combining RAP and RFM, 'Adv+GAPS' detector have higher detection performance than 'Adv+RAP' detector on four adversarial attack [9], [25], [26], [37]. From the above analysis, we conclude that under the premise of using RAP, RFM can guide detectors to focus on classifying sensitive regions.

C. Training without Any adv-faces

In this part, we propose a new problem setting: **Training detectors without any real adv-faces while enabling detectors to efficiently detect them.** For this new setting, we use our GAPS to train detectors using sadv-faces. Three types of simulated adversarial perturbations we used for detectors training are shown in Fig 4. As shown in Table II, we find that 'Real+GAPS' and 'Real+RAP' detectors which trained without any real adv-faces achieve higher performance than detectors trained with adv-faces generated by FGSM. In particular, 'Real+RAP' increases TIFGSM detection ACC to 98.2%, indicating that our method can effectively guide detectors to learn more general adversarial perturbation features and improve the generalizability of detectors. From the above we can conclude that it is feasible to train detectors using GAPS or RAP without any real adv-faces.

TABLE III
TIPIM DETECTION PERFORMANCE OF SIX DIFFERENT DATA SETTING DETECTORS, THE NUMBERS IN THE TABLE REPRESENT DETECTORS DETECTION ACC.

Detector	TIPIM [27]
Adv	77.4425
Adv + RFM	76.0561
Adv + RAP	96.8596
Adv + GAPS	96.8596
Real + RAP	**99.5966**
Real + GAPS	97.6722

D. Defense against possible future attacks

Through the analysis in subsection 3.1, we found that adversarial perturbations generated by seven adversarial attacks have fixed patterns and we speculate that adversarial perturbations generated by possible future attacks may also conform to these patterns. To validate our speculation, we additionally choose the SOTA gradient-based adversarial attack TIPIM [27] to test detectors performance. From Table III, we can clearly conclude GAPS can effectively improve TIPIM detection ACC, detectors trained only with RFM cannot achieve satisfactory performance. In the meantime, 'Real+RAP' and 'Real+GAPS' detectors achieve higher performance than other detectors, indicating that detectors trained without any real adv-faces have efficient defense ability against possible future attacks. The above experiments prove that TIPIM perturbation also conforms to patterns we previously assumed and our GAPS can guide detectors to defend possible future attacks.

VI. Conclusion

Adversarial faces (adv-faces) hampered the reliability of face recognition systems (FRSs) and rippled to financial industry, however previous adv-face detection method require either pre-trained samples from known attack methods or accessing to the protected FRSs and hence difficult to be applied to practical scenarios. Considering these deficiencies, we propose a FRS-agnostic and attack-agnostic method for general adv-face detection, General Adversarial Perturbation Simulating (GAPS), which is able to defend any unknown FRSs against any unseen adversarial attacks while do not modify or access to the FRS. GAPS effectively simulates fixed patterns of adversarial perturbations and guide detectors to focus on classifying sensitive regions while enabling vanilla CNNs based detectors to efficiently detect adv-faces. Trained with only real face images and sadv-faces, detectors are able to achieve satisfactory detection performances under the guidance of GAPS. Extensive experiments on LFW and CASIA-WebFace datasets demonstrate the effectiveness of our method on detecting adv-faces generated by various of attack methods.

References

[1] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Advances in neural information processing systems*, vol. 25, 2012.

[2] Z. Li, Y. Shi, H. Ling, J. Chen, Q. Wang, and F. Zhou, "Reliability exploration with self-ensemble learning for domain adaptive person re-identification," 2022.

[3] Y. Shi, H. Ling, L. Wu, J. Shen, and P. Li, "Learning refined attribute-aligned network with attribute selection for person re-identification," *Neurocomputing*, vol. 402, pp. 124–133, 2020.

[4] H. Ling, Z. Wang, P. Li, Y. Shi, J. Chen, and F. Zou, "Improving person re-identification by multi-task learning," *Neurocomputing*, vol. 347, pp. 109–118, 2019.

[5] 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.

[6] H. Ling, J. Wu, J. Huang, J. Chen, and P. Li, "Attention-based convolutional neural network for deep face recognition," *Multimedia Tools and Applications*, vol. 79, no. 9, pp. 5595–5616, 2020.

[7] H. Ling, J. Wu, L. Wu, J. Huang, J. Chen, and P. Li, "Self residual attention network for deep face recognition," *IEEE Access*, vol. 7, pp. 55 159–55 168, 2019.

[8] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus, "Intriguing properties of neural networks," *arXiv preprint arXiv:1312.6199*, 2013.

[9] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," *arXiv preprint arXiv:1412.6572*, 2014.

[10] G. Tao, S. Ma, Y. Liu, and X. Zhang, "Attacks meet interpretability: Attribute-steered detection of adversarial samples," in *Advances in Neural Information Processing Systems*, S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, Eds., vol. 31. Curran Associates, Inc., 2018.

[11] Z. Deng, X. Yang, S. Xu, H. Su, and J. Zhu, "Libre: A practical bayesian approach to adversarial detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 972–982.

[12] F. V. Massoli, F. Carrara, G. Amato, and F. Falchi, "Detection of face recognition adversarial attacks," *Computer Vision and Image Understanding*, vol. 202, p. 103103, 2021.

[13] C. Wang and W. Deng, "Representative forgery mining for fake face detection," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 14 923–14 932.

[14] G. B. Huang, M. Mattar, T. Berg, and E. Learned-Miller, "Labeled faces in the wild: A database forstudying face recognition in unconstrained environments," in *Workshop on faces in'Real-Life'Images: detection, alignment, and recognition*, 2008.

[15] D. Yi, Z. Lei, S. Liao, and S. Z. Li, "Learning face representation from scratch," *arXiv preprint arXiv:1411.7923*, 2014.

[16] Z. Gong, W. Wang, and W.-S. Ku, "Adversarial and clean data are not twins," *arXiv preprint arXiv:1704.04960*, 2017.

[17] S. Liang, Y. Li, and R. Srikant, "Enhancing the reliability of out-of-distribution image detection in neural networks," *arXiv preprint arXiv:1706.02690*, 2017.

[18] J. Liu, W. Zhang, Y. Zhang, D. Hou, Y. Liu, H. Zha, and N. Yu, "Detection based defense against adversarial examples from the steganalysis point of view," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 4825–4834.

[19] A. Agarwal, R. Singh, M. Vatsa, and N. Ratha, "Are image-agnostic universal adversarial perturbations for face recognition difficult to detect?" in *2018 IEEE 9th International Conference on Biometrics Theory, Applications and Systems (BTAS)*. IEEE, 2018, pp. 1–7.

[20] A. Goel, A. Singh, A. Agarwal, M. Vatsa, and R. Singh, "Smartbox: Benchmarking adversarial detection and mitigation algorithms for face recognition," in *2018 IEEE 9th International Conference on Biometrics Theory, Applications and Systems (BTAS)*. IEEE, 2018, pp. 1–7.

[21] G. Goswami, A. Agarwal, N. Ratha, R. Singh, and M. Vatsa, "Detecting and mitigating adversarial perturbations for robust face recognition," *International Journal of Computer Vision*, vol. 127, no. 6, pp. 719–742, 2019.

[22] A. Agarwal, R. Singh, M. Vatsa, and N. Ratha, "Image transformation-based defense against adversarial perturbation on deep learning models," *IEEE Transactions on Dependable and Secure Computing*, vol. 18, no. 5, pp. 2106–2121, 2020.

[23] D. Deb, X. Liu, and A. K. Jain, "Faceguard: A self-supervised defense against adversarial face images," *arXiv preprint arXiv:2011.14218*, 2020.

[24] A. Kurakin, I. J. Goodfellow, and S. Bengio, "Adversarial examples in the physical world," in *Artificial intelligence safety and security*. Chapman and Hall/CRC, 2018, pp. 99–112.

[25] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu, "Towards deep learning models resistant to adversarial attacks," *arXiv preprint arXiv:1706.06083*, 2017.

[26] Y. Dong, F. Liao, T. Pang, H. Su, J. Zhu, X. Hu, and J. Li, "Boosting adversarial attacks with momentum," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 9185–9193.

[27] X. Yang, Y. Dong, T. Pang, H. Su, J. Zhu, Y. Chen, and H. Xue, "Towards face encryption by generating adversarial identity masks," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 3897–3907.

[28] W. Xu, D. Evans, and Y. Qi, "Feature squeezing: Detecting adversarial examples in deep neural networks," *arXiv preprint arXiv:1704.01155*, 2017.

[29] G. K. Dziugaite, Z. Ghahramani, and D. M. Roy, "A study of the effect of jpg compression on adversarial images," *arXiv preprint arXiv:1608.00853*, 2016.

[30] A. Shafahi, M. Najibi, Z. Xu, J. Dickerson, L. S. Davis, and T. Goldstein, "Universal adversarial training," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 04, 2020, pp. 5636–5643.

[31] S. Sankaranarayanan, A. Jain, R. Chellappa, and S. N. Lim, "Regularizing deep networks using efficient layerwise adversarial training," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 32, no. 1, 2018.

[32] H. Dang, F. Liu, J. Stehouwer, X. Liu, and A. K. Jain, "On the detection of digital face manipulation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern recognition*, 2020, pp. 5781–5790.

[33] Z. Zhong, L. Zheng, G. Kang, S. Li, and Y. Yang, "Random erasing data augmentation," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 34, no. 07, 2020, pp. 13 001–13 008.

[34] H. Kim, "Torchattacks: A pytorch repository for adversarial attacks," *arXiv preprint arXiv:2010.01950*, 2020.

[35] F. Tramèr, A. Kurakin, N. Papernot, I. Goodfellow, D. Boneh, and P. McDaniel, "Ensemble adversarial training: Attacks and defenses," *arXiv preprint arXiv:1705.07204*, 2017.

[36] Y. Dong, T. Pang, H. Su, and J. Zhu, "Evading defenses to transferable adversarial examples by translation-invariant attacks," in *Proceedings of*

the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 4312–4321.

[37] C. Xie, Z. Zhang, Y. Zhou, S. Bai, J. Wang, Z. Ren, and A. L. Yuille, "Improving transferability of adversarial examples with input diversity," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 2730–2739.

[38] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 1251–1258.

[39] A. KingaD, "A method for stochastic optimization," *Anon. InternationalConferenceon Learning Representations. SanDego: ICLR*, 2015.