Deceiving the Protector: Fooling Face Presentation Attack Detection Algorithms

Akshay Agarwal, Akarsha Sehwag, Mayank Vatsa, and Richa Singh
{akshaya, akarsha15010, mayank, rsingh}@iiitd.ac.in
IIIT-Delhi, India

Abstract

Face recognition systems are vulnerable to presentation attacks such as replay and 3D masks. In the literature, several presentation attack detection (PAD) algorithms are developed to address this problem. However, for the first time in the literature, this paper showcases that it is possible to "fool" the PAD algorithms using adversarial perturbations. The proposed perturbation approach attacks the presentation attack detection algorithms at the PAD feature level via transformation of features from one class (attack class) to another (real class). The PAD feature tampering network utilizes convolutional autoencoder to learn the perturbations. The proposed algorithm is evaluated with respect to CNN and local binary pattern (LBP) based PAD algorithms. Experiments on three databases, Replay, SMAD, and Face Morph, showcase that the proposed approach increases the equal error rate of PAD algorithms by at least two times. For instance, on the SMAD database, PAD equal error rate (EER) of 20.1% is increased to 55.7% after attacking the PAD algorithm.

1. Introduction

Face recognition (FR) systems are currently being used in a variety of applications including surveillance, secure access, and mobile banking. While the usage of face recognition technology provides additional security compared to pins and passwords, the security of these systems itself is also of paramount importance. Researchers have demonstrated that face recognition systems can be circumvented by using the photo of genuine users [12], or by wearing a 3D mask [20]. In the real world, there had been instances of attacks where robbers wearing 3D silicone masks tried to fool the face recognition system. The presence of these attacks shows the vulnerability of face recognition systems.

The attack at the sensor level (i.e., at data acquisition level) performed using photo and mask are popularly known as presentation attacks. To secure the system from sensor and image level attacks, researchers have proposed various attack detection algorithms either by extracting the hand-crafted features such as texture or motion cues or deep learning features. The algorithms developed to secure the system from presentation attacks are known as *presentation attack detection (PAD)* algorithms. While PAD algorithms are able to secure the system from these attacks, these algorithm might also be sensitive towards PAD feature tampering. In other words, an attacker can attack the PAD feature extraction and/or decision modules and the altered decision of this sub-system can affect the performance of the whole face recognition system. As a first ever attempt, this research explores the possibility of tampering the PAD algorithms themselves by attacking PAD algorithm at the feature extraction level.

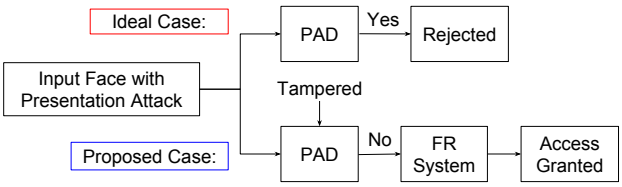


Figure 1: Diagram explaining the motivation of the proposed research.

In related studies, it has been shown that the deep learning algorithms are susceptible to small changes in the pixel domain [3, 8]. Szegedy et al. [24] observe that small manipulations in the pixel values can lead to its misclassification by deep learning algorithms. Similarly, Goswami et al. [15] also performed various kinds of image and face level attacks to fool the deep learning based face recognition algorithms. However, most of the existing adversarial attacks fool the deep classifiers at the image level only. While image alterations can be used to deceive a PAD algorithm, it may also affect the performance of the face recognition algorithm. On the other hand, attacking PAD algorithm at the PAD feature level (not at the image level) may ensure that the input to the face matcher is not perturbed but decision of PAD algorithm is altered (from *attacked* to *real*). This would ensure that the input image matches the targeted identity and can deceive the presentation attack detector as well.

https://tinyurl.com/yb2dxsc9, https://tinyurl.com/y9gn4tmk

Figure 1 shows the motivation of the proposed feature tampering algorithm. As mentioned earlier, an attacker may attack/tamper the PAD module and change the prediction so that the face recognition (FR) module performs the recognition task and unauthorized access may be granted. In this paper, we propose a convolutional auto-encoder based network to *fool* both handcrafted and representation learning based presentation attack detector so that the face recognition module can also be fooled. Extensive experiments on multiple databases with both, 'inter' and 'intra' attacks (or database) showcase the strength of the proposed PAD feature level attack in fooling presentation attack detectors. We have also prepared the digital presentation attack database, referred as Face Morph, containing 70 morphed videos.

2. Literature Review

A lot of work has been done to mislead the face recognition system at sensor level using photo, replay of video, or 3D masks as well as image level using morphing or retouching. Similarly, various algorithms have been proposed to detect the presentation attacks on face recognition system. However, to the best of our knowledge, attack on PAD algorithm has not been performed yet.

PAD algorithms can be classified into texture, motion, and representation learning based approaches. Texture based algorithms such as Gaussian [27], LBP [19, 23], Gabor, Histogram of Oriented Gradients (HOG), Wavelet+ Haralick [1] are the most popular and provide state-of-the-art (SOTA) detection performances across various types of databases. The combination of texture and motion also shows tremendous performance in detecting presentation attacks [5, 13].

The success of deep learning in object detection and recognition have motivated the researchers to explore it for presentation attack detection. Menotti et al. [21], Li et al. [17], De Souza et al. [11], Lin et al. [18], and Manjani et al. [20] have proposed CNN and deep dictionary algorithms to detect various kinds of presentation attacks including silicone mask attacks. The details of the existing PAD algorithm can be found in [16]. In addition, Agarwal et al. [2] and Raghavendra et al. [22] have proposed texture-based classification approaches for detecting digital morph attack on face recognition. While not directly related to face presentation attack literature, researchers have also shown that adversarial noise can be learned using deep neural networks or hand-crafted to fool face recognition algorithms. Literature related to adversarial noise detection algorithms can be found in [4, 14].

3. Proposed Algorithm

The generic diagram of the proposed feature tampering network is shown in Figure 2. The proposed PAD feature tampering based attack aims to fool the presentation attack

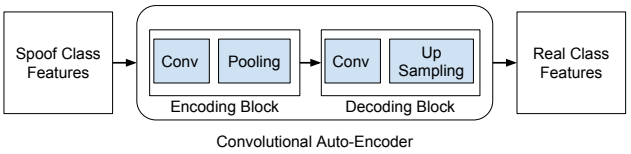


Figure 2: Generic diagram of the proposed feature tampering using convolutional auto-encoder.

detector. In other words, the objective is to learn a mapping between the feature maps of *real* and *spoof*, which are of the same dimension. For this task, as shown in Figure 3, a convolutional auto-encoder (CAE) network is trained on the PAD features.

Let I be the input feature of size $x \times y$ with depth D' and F be the CAE filters of size $m \times n$ with depth D.

$$\begin{aligned} I &= I_1, \cdots, I_{D'}, \\ F &= F_1, \cdots, F_D \end{aligned} \tag{1}$$

The output of the convolution step is defined as:

$$O(i,j) = \sum_{d=1}^{D,D'} \sum_{a=0}^{m-1} \sum_{b=0}^{n-1} F_d(a,b) I_d(i+a, j+b) \tag{2}$$

The convolutional feature maps after non-linearity are represented as: $z_m = \sigma(O)$. The output maps produced by non-linear activation function (σ) followed by max pooling layers are referred as encoding feature maps of CAE. The decoding steps are also convolution followed by up-sampling so that the output must be of the same size as the input. The decoding step of CAE on the encoded feature maps $Z = \{z_{i=1}\}^n$ is:

$$\tilde{I} = \varphi(Z * F) \tag{3}$$

where, φ and $*$ represents the up-sampling and convolution function, respectively. The encoding-decoding in terms of dimensions can be mathematically written as:

$$dim(I) = dim(decode(encode(I))) \tag{4}$$

While the features of spoof data are used as input (I) of the network, features of the real data are used for mapping at the output ($\tilde{I}$) of the network, and vice-versa. The mean squared loss is minimized between the expected feature vector generated using the network and feature vectors of real data. The encoding block of the network contains 2 convolutional layers each followed by max pooling. Similarly, the decoding block contains 2 convolutional layers, each followed by an upsampling layer. The task of the up-sampling layer is to increase the size of the convolutional feature map so that the output can map to the input feature dimension. Each convolutional layer of the encoding and decoding blocks contain 32 filters of size 3×3. The network is trained using stochastic gradient descent with mean

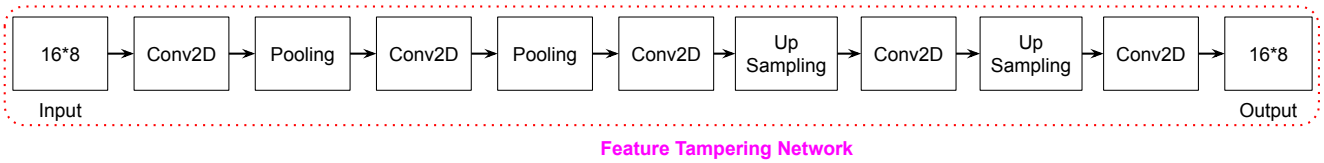


Figure 3: Figure illustrating the proposed feature tampering algorithm with CNN based PAD.

squared loss as the objective function. The momentum and learning rate of the network in training is set to 0.95 and 10^{-5}, respectively. The assumption of the proposed attack is the knowledge of features used for PAD algorithm.

4. Databases and Evaluation Setup

To demonstrate the effectiveness of the proposed PAD feature tampering network, we have used two PAD algorithms and multiple face databases.

4.1. PAD Algorithms

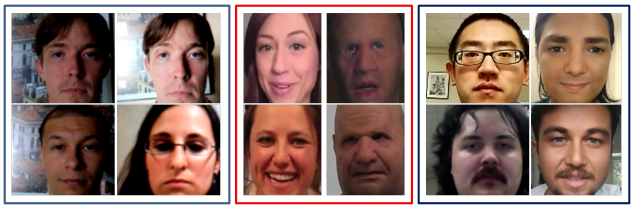
As discussed in the literature review, both handcrafted as well as representation learning based approaches are used for presentation attack detection. Therefore, we perform the experiments with two algorithms: (1) CNN-based and (2) local binary pattern-based.

CNN-based PAD Algorithm: The CNN architecture used in this research is inspired from the architecture proposed in Chatfield et al. [7] and Zeiler and Fergus [26]. The network is trained on ILSVRC-2012 database with momentum using gradient descent optimization. The hyper parameters of the network such as momentum and learning rate are set to 0.9 and 10^{-2}, respectively. The learning rate decreases by a factor of 10 when the decrease in the error-rate on the validation set is very low. To limit the computational requirement, the dimension of the last fully connected layer is set to 128. The 128 dimensional 1D feature vector computed using CNN is converted into the shape of 16×8 to train the proposed PAD feature tampering network. The support vector machine (SVM) [10] classifier is trained for presentation attack detection on CNN features.

LBP-based PAD Algorithm [9, 19]: The traditional LBP histogram features are computed both from the real and spoof images. The LBP image is formed by comparing the center pixel of each 3×3 grid of the image from its 8 neighborhood pixels. The final integer value at the center location is computed by assigning the label from 0 to 2^8-1.

4.2. Databases

We have performed the experiments with four different kinds of attacks: 2D photo, 2D video, silicone mask, and digital morphing. Therefore, the databases used are: (i) Replay-Attack [9], SMAD [20], and the proposed digital face morph database. Figure 4 shows sample images from each category of the presentation attack databases, i.e., physical to digital.



(a) Real and 2D Attack Samples (b) Real and 3D Attack Samples (c) Real and Digital Attack Samples

Figure 4: Samples images from all three databases used. (a) Replay-Attack [9], (b) SMAD [20], and (c) MSU-MFSD [25] and proposed face morph database. First column in each are the real samples and last column is the attack samples.

Replay-Attack Database: In total, the database contains 1,200 videos out of which 200 are real access videos, and 1,000 are print, photo, and replay attack videos. The database is divided into three subsets: train, validation, and test as described in the original paper. The videos are collected using both high-def and mobile camera and assembled using fixed and hand-held medium. Hence, the results are reported individually on both the mediums, i.e., Fixed and Hand-held.

SMAD Database: It consists of the real and 3D silicone mask attack videos acquired from online sources. It contains 65 real and 65 mask attack videos. In this paper, the database has been divided into training and testing sets, where the training set has 40% videos of both classes. The remaining 60% videos of both the classes are used for evaluation.

Face Morph Database: The two above mentioned databases are physical presentation attack databases: where the attackers wear or show the attacking medium in front of the camera. The other possible attack on face recognition system is through morphing, swapping, and retouching [2,6]. To cover this spectrum of presentation attack, we have prepared the face morph database using Snapchat mobile application. In total, 70 face morphed videos are collected using Snapchat in both constrainted and unconstrainted environment. In place of collecting the real videos, we took real access videos from MSU-MFSD database [25] which contains 70 real videos captured from 35 subjects using mo-

bile devices including Android phone. The real subset of MSU database is divided into training and testing containing videos of 15 subjects in training and videos of 20 subjects in testing. Similarly, we divided the morph videos into training and testing sets containing 30 and 40 videos, respectively.

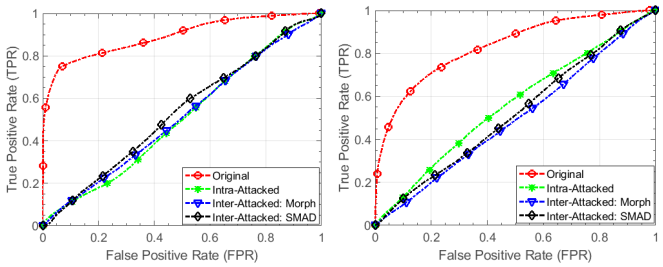
4.3. Performance Metrics

The problem of presentation attack detection is a binary classification problem. Hence, the proposed algorithm can have two kinds of errors referred to as false acceptance rate (FAR) and false rejection rate (FRR). The FAR and FRR metrics corresponds to attack presentation attack classification error rate (APCER) and bona-fide presentation attack classification error rate (BPCER), respectively. In this paper, the performance of the PAD algorithms and feature tampering algorithm are reported in terms of equal error rate (EER). EER is defined as the point on receiver operating characteristics (ROC) curve where FAR is equal to FRR. The higher the difference between the EER of original tampered features, more effective is the tampering algorithm.

5. Experimental Results and Analysis

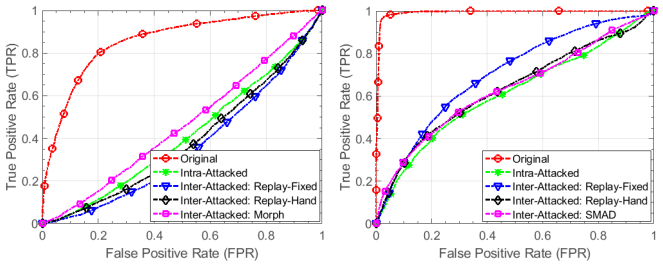
As stated earlier, this is the first work performing an attack to fool the face presentation attack detector. The features are transformed in such a way that an attack performed at the sensor level can bypass the PAD algorithms. Two kinds of experiments are performed: intra-attack and inter-attack. The intra-attack experiments can be defined as the setup in which the feature tampering network is trained using the same database as the testing database. On the other hand, the inter-attack experiments are the ones in which the tampering network is trained using one database (such as Replay-Attack), but the tampering is performed on the features computed from other databases (such as SMAD). The inter-attack experiments are essential for real-world evaluation where the attacking medium might be new at the time of training. Similarly, the sensitivity of different SVM kernels is also evaluated against feature tampering. ROC curves for the CNN based presentation attack detection on all three databases: Replay-Attack, SMAD, and Morph are shown in Figures 5 and 6, respectively. The red curve which is on the original PAD features depicts the higher detection accuracy (i.e. lower EER). Other curves obtained on the tampered features shows approximately random behavior of face PAD algorithm. Next, we discuss the analysis related to intra-attack experiments followed by the findings related to inter-attack experiments.

Intra-Attack Results: The results of this experiment are reported in Table 1. The presentation attack detector is first trained using the CNN features computed on the training set of each database. With original (un-perturbed) images, the trained PAD algorithm with linear kernel yields an equal



(a) Fixed Set (b) Hand-held Set

Figure 5: ROC curve of face presentation attack detection on original, intra-attack, and inter-attack tampered features on Replay-Attack database.



(a) SMAD (b) Morph

Figure 6: ROC curve of face presentation attack detection on original, intra-attack, and inter-attack tampered features on SMAD and Face Morph database.

error rate (EER) of 25.1% and 19.3% on hand-held and fixed set of Replay-Attack database, respectively. Similarly, on Silicone Mask Attack database (i.e., SMAD) and Digital Attack database (i.e., face morph), the SVM detector with *linear* kernel yields lowest EER value of 20.1% and 2.8%, respectively on the original feature set. On 2D attacks, the SVM with *RBF* kernel yields the lowest EER value whereas on silicone and digital morph attack *linear* kernel shows the best performance. Since some PAD algorithms are handcrafted in nature, we also performed some of the experiments with LBP based PAD algorithms. The results related to face presentation attack detection using original and tampered LBP features are summarized in Table 2. The results show that both deep CNN features and handcrafted features based PAD algorithms are not robust to tampering.

With feature tampering the EER(%) on digital face morph database increased by more than 14, 10, and 16 times with linear, polynomial, and RBF kernel-based SVM with CNN features, respectively. Similar to the Morph digital attack database, PAD algorithm trained for 2D and 3D presentation attack databases shows sensitivity towards feature tampering. EER (%) on SMAD database increases to 55.7% from 20.1% (linear SVM) and on the Replay-Attack database (Hand-held), it increases to 45.4% from 21.2% (polynomial SVM).

Table 1: CNN-PAD performance (EER%) with original and proposed CAE tampered features.

Database	Kernel	Features		Difference
		Original	Tampered	
Replay-Attack (Hand-held)	Linear	25.1	45.2	20.1
	Polynomial	21.2	45.4	24.2
	RBF	**21.4**	**44.8**	23.4
Replay-Attack (Fixed)	Linear	19.3	50.1	30.8
	Polynomial	19.3	47.7	28.4
	RBF	**16.7**	**49.8**	33.7
SMAD	Linear	**20.1**	**55.7**	35.6
	Polynomial	20.3	57.5	37.2
	RBF	23.7	58.1	34.4
Face Morph	Linear	**2.8**	**41.5**	38.7
	Polynomial	3.5	38.0	34.5
	RBF	4.0	65.5	61.5

Table 2: LBP-PAD performance (EER%) with original and proposed CAE tampered features.

Database	Kernel	Features		Difference
		Original	Tampered	
SMAD	Linear	**25.1**	**46.9**	21.8
	Polynomial	30.8	45.9	15.1
	RBF	32.9	43.6	10.7
Face Morph	Linear	0.0	86.1	86.1
	Polynomial	0.1	73.4	73.3
	RBF	**0.0**	**35.7**	35.7

Table 3: CNN-PAD performance (EER%) when CAE is trained using SMAD.

Database	Kernel	Features	
		Original	Tampered
Replay-Attack (Hand-held)	Linear	25.1	49.4
	Polynomial	**21.2**	**48.3**
	RBF	21.4	49.2
Replay-Attack (Fixed)	Linear	19.3	46.8
	Polynomial	19.3	46.4
	RBF	**16.7**	**46.8**
Face Morph	Linear	**2.8**	**39.9**
	Polynomial	3.5	39.0
	RBF	4.0	67.9

Table 4: CNN-PAD performance (EER%) when CAE is trained using Face Morph database.

Database	Kernel	Features	
		Original	Tampered
Replay-Attack (Hand-held)	Linear	25.1	50.5
	Polynomial	**21.2**	**49.5**
	RBF	21.4	52.0
Replay-Attack (Fixed)	Linear	19.3	49.6
	Polynomial	19.3	49.3
	RBF	**16.7**	**49.8**
SMAD	Linear	**20.1**	**52.6**
	Polynomial	20.3	52.7
	RBF	23.7	53.3

Inter-Attack Results: Tables 3, 4, 5, and 6 show the presentation attack detection performance for inter-attack experiments with CNN-PAD. The results in Table 3 correspond to when the feature tampering network is trained on 3D mask attack database, and features of 2D attack and Morph

Table 5: CNN-PAD performance (EER%) when CAE is trained using Replay-Attack (Fixed).

Database	Kernel	Features	
		Original	Tampered
SMAD	Linear	**20.1**	**59.6**
	Polynomial	20.3	56.6
	RBF	23.7	53.2
Face Morph	Linear	**2.8**	**34.9**
	Polynomial	3.5	35.3
	RBF	4.0	62.2

Table 6: CNN-PAD performance (EER%) when CAE is trained using Replay-Attack (Hand-held).

Database	Kernel	Features	
		Original	Tampered
SMAD	Linear	**20.1**	**58.3**
	Polynomial	20.3	57.8
	RBF	23.7	56.2
Face Morph	Linear	**2.8**	**40.2**
	Polynomial	3.5	39.8
	RBF	4.0	56.5

database are tampered. Even when the network is trained on a completely unseen attack, the feature tampering is able to degrade the performance of the presentation attack detector. For example, as shown in Table 3, when the convolutional-autoencoder is trained using the training set of SMAD but tested on Face Morph database, the EER increases from 2.8% to 41.5% (for *linear* SVM). The performance degradation of the PAD algorithm under inter-attack scenario shows the real-world application of the proposed feature level attack. The performance degradation across each SVM kernel on the hand-held set of Replay-Attack database is even higher under inter-attack network training in comparison to intra-attack learning. Similar susceptibility of the PAD algorithm is observed on SMAD and Morph databases when the attacking network is trained on fixed and hand-held sets of the Replay-Attack database.

Similarly, when the feature tampering algorithm is trained on digital attack and tested on physical attack, the PAD algorithm is not able to maintain the detection performance (shown in Table 4). On SMAD database, the EER increases approximately by the same percentage as intra-attack feature tampering. The performance degradation of the PAD algorithm across inter and intra attacks on all the databases shows the generalizability of the proposed tampering algorithm. We have also noticed that the distribution of the feature vectors changes significantly after alteration, thus leads to mis-classification by the PAD algorithm.

6. Conclusion

In this paper, we showcase that the *protector can be deceived*. The proposed feature tampering approach utilizes

convolutional auto-encoder based network to learn the perturbation for fooling face PAD algorithms. The proposed attack is evaluated across multiple attacks and databases including 2D photo, silicone mask, and digital morphing. The proposed approach, both in intra and inter attack scenarios, shows the susceptibility of the PAD algorithms. Apart from this, a new digital morph database using the Snapchat mobile application is prepared. In future, efforts can be made to increase the robustness of the face presentation attack detection algorithms against PAD feature tampering.

7. Acknowledgement

A. Agarwal is partly supported by Visvesvaraya PhD Fellowship, and M. Vatsa and R. Singh are partly supported from the Infosys Center for AI at IIIT-Delhi. M. Vatsa is also partially supported by the Department of Science and Technology, Government of India through Swarnajayanti Fellowship.

References

[1] A. Agarwal, R. Singh, and M. Vatsa. Face anti-spoofing using haralick features. In *IEEE BTAS*, pages 1–6, 2016.

[2] A. Agarwal, R. Singh, M. Vatsa, and A. Noore. Swapped! Digital face presentation attack detection via weighted local magnitude pattern. In *IJCB*, pages 659–665, 2017.

[3] A. Agarwal, R. Singh, M. Vatsa, and N. Ratha. Are image-agnostic universal adversarial perturbations for face recognition difficult to detect? *IEEE BTAS*, 2018.

[4] N. Akhtar and A. Mian. Threat of adversarial attacks on deep learning in computer vision: A survey. *IEEE Access*, 6:14410–14430, 2018.

[5] S. Bharadwaj, T. I. Dhamecha, M. Vatsa, and R. Singh. Computationally efficient face spoofing detection with motion magnification. In *IEEE CVPRW*, pages 105–110, 2013.

[6] A. Bharati, M. Vatsa, R. Singh, K. W. Bowyer, and X. Tong. Demography-based facial retouching detection using subclass supervised sparse autoencoder. In *IEEE IJCB*, pages 474–482, 2017.

[7] K. Chatfield, K. Simonyan, A. Vedaldi, and A. Zisserman. Return of the devil in the details: Delving deep into convolutional nets. In *BMVC*, 2014.

[8] S. Chhabra, R. Singh, M. Vatsa, and G. Gupta. Anonymizing k-facial attributes via adversarial perturbations. *IJCAI*, 2018.

[9] I. Chingovska, A. Anjos, and S. Marcel. On the effectiveness of local binary patterns in face anti-spoofing. In *BIOSIG*, number EPFL-CONF-192369, 2012.

[10] C. Cortes and V. Vapnik. Support vector machine. *Machine Learning*, 20(3):273–297, 1995.

[11] G. B. de Souza, J. P. Papa, and A. N. Marana. On the learning of deep local features for robust face spoofing detection. *arXiv preprint arXiv:1806.07492*, 2018.

[12] N. M. Duc and B. Q. Minh. Your face is not your password face authentication bypassing lenovo–asus–toshiba. *Black Hat*, 2009.

[13] L. Feng, L.-M. Po, Y. Li, X. Xu, F. Yuan, T. C.-H. Cheung, and K.-W. Cheung. Integration of image quality and motion cues for face anti-spoofing: A neural network approach. *JVCIR*, 38:451–460, 2016.

[14] G. Goswami, A. Agarwal, N. Ratha, R. Singh, and M. Vatsa. Detecting and mitigating adversarial perturbations for robust face recognition. *IJCV*, 2019.

[15] G. Goswami, N. Ratha, A. Agarwal, R. Singh, and M. Vatsa. Unravelling robustness of deep learning based face recognition against adversarial attacks. *AAAI*, 2018.

[16] J. Hernandez-Ortega, J. Fierrez, A. Morales, and J. Galbally. Introduction to face presentation attack detection. In *Handbook of Biometric Anti-Spoofing*, pages 187–206. Springer, 2019.

[17] H. Li, P. He, S. Wang, A. Rocha, X. Jiang, and A. C. Kot. Learning generalized deep feature representation for face anti-spoofing. *IEEE TIFS*, 13(10):2639–2652, 2018.

[18] C. Lin, Z. Liao, P. Zhou, J. Hu, and B. Ni. Live face verification with multiple instantialized local homographic parameterization. In *IJCAI*, pages 814–820, 2018.

[19] J. Määttä, A. Hadid, and M. Pietikäinen. Face spoofing detection from single images using micro-texture analysis. In *IJCB*, pages 1–7, 2011.

[20] I. Manjani, S. Tariyal, M. Vatsa, R. Singh, and A. Majumdar. Detecting silicone mask-based presentation attack via deep dictionary learning. *IEEE TIFS*, 12(7):1713–1723, 2017.

[21] D. Menotti, G. Chiachia, A. Pinto, W. R. Schwartz, H. Pedrini, A. X. Falcao, and A. Rocha. Deep representations for iris, face, and fingerprint spoofing detection. *IEEE TIFS*, 10(4):864–879, 2015.

[22] R. Raghavendra, K. B. Raja, and C. Busch. Detecting morphed face images. In *IEEE BTAS*, pages 1–7, 2016.

[23] T. A. Siddiqui, S. Bharadwaj, T. I. Dhamecha, A. Agarwal, M. Vatsa, R. Singh, and N. Ratha. Face anti-spoofing with multifeature videolet aggregation. In *ICPR*, pages 1035–1040, 2016.

[24] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus. Intriguing properties of neural networks. *ICLR*, 2014.

[25] D. Wen, H. Han, and A. K. Jain. Face spoof detection with image distortion analysis. *IEEE TIFS*, 10(4):746–761, 2015.

[26] M. D. Zeiler and R. Fergus. Visualizing and understanding convolutional networks. In *ECCV*, pages 818–833. Springer, 2014.

[27] Z. Zhang, J. Yan, S. Liu, Z. Lei, D. Yi, and S. Z. Li. A face antispoofing database with diverse attacks. In *IAPR ICB*, pages 26–31, 2012.