Salience-Aware Face Presentation Attack Detection via Deep Reinforcement Learning

Bingyao Yu, Jiwen Lu, *Senior Member, IEEE,* Xiu Li, and Jie Zhou *Senior Member, IEEE*

***Abstract*—In this paper, we propose a salience-aware face presentation attack detection (SAFPAD) approach, which takes advantage of deep reinforcement learning to exploit the salient local part information in face images. Most existing deep face presentation attack detection approaches extract features from the entire image or several fixed regions. However, the discriminative information beneficial for presentation attack detection is unevenly distributed in the image due to the illumination and presentation attack instrument variation, so treating all regions equally fails to highlight the most discriminative information which is important for more accurate and robust face presentation attack detection. To address this, we propose to identify the discriminative salient parts using deep reinforcement learning and focus on them to alleviate the adverse effects of redundant information in the face images. We fuse the high-level features and the local features which guide the policy network to exploit discriminative patches and assist the classification network to predict more accurate results. We jointly train the SAFPAD model with deep reinforcement learning to generate salient locations. Extensive experiments on five public datasets demonstrate that our approach achieves very competitive performance due to the concentrated employment of salient local information.**

***Index Terms*—Face presentation attack detection, deep reinforcement learning, multiscale feature fusion.**

I. Introduction

FACE recognition systems have been widely used in multiple applications, such as access control, mobile payment, and security check [1], [2]. Despite the rapid development and increasing popularity of face recognition techniques, many face recognition systems turn out to be vulnerable to face spoofs and presentation attacks (PA) [2]–[5]. Nowadays, the employment of face information for person identification brings convenience to users, yet it also attracts the attention of ill-intentioned attackers at the same time. Utilizing the highly developed social network and internet technology, attackers are able to obtain users' photos and even videos without much effort. To gain an illegal authentication identity, an attacker can print a photo on paper (i.e., print attack), play a digital video (i.e., replay attack), or wear a 3D mask (i.e., mask attack) to deceive the face recognition system [6], [7]. Consequently, the demand for effective face presentation attack detection methods, which serve as the shield of face recognition systems, is well justified.

Bingyao Yu and Xiu Li are with the Tsinghua Shenzhen International Graduate School, Shenzhen, 518055, China. E-mail: yby18@mails.tsinghua.edu.cn; li.xiu@sz.tsinghua.edu.cn.

Jiwen Lu and Jie Zhou are with the Department of Automation, Beijing Research Center for Information Science and Technology (BNRist), Tsinghua University, Beijing, 100084, China. E-mail: lujiwen@tsinghua.edu.cn; jzhou@tsinghua.edu.cn.

Corresponding author: Xiu Li

Face presentation attack detection is of great significance to the security of face recognition systems, but it remains very challenging and unsolved, in spite of a variety of attempts to extract discriminative features for effective face presentation attack detection [8], [9]. Existing face presentation attack detection methods can be mainly divided into three classes: texture-based methods, temporal-based methods and hardware-based methods [10]–[14]. Temporal-based methods take as input face videos to perform presentation attack detection, which need more computing resources and are not convenient in practical applications. Hardware-based methods have achieved reasonably good performance by utilizing multi-modal data, but they demand various sensors, which unavoidably brings high costs to applications in real life [15]. Therefore, we focus on the first category of texture-based methods, which are able to perform face presentation attack detection using only a single RGB image. The early texture-based face presentation attack detection methods combine the hand-crafted features and SVM (Support Vector Machine) to analyze the intrinsic differences between bona fide faces and attack presentations [16]–[18]. With the development of deep learning, recent texture-based methods employ CNNs to learn representations under the supervision of softmax loss function [19]–[21] and achieve promising results. The majority of existing texture-based methods extract features from the entire input image and neglect the unequal salience of different regions. However, the information of an input image consists of a number of discriminative components with different levels of discrimination. For example, as demonstrated in Fig. 1, the discriminative information beneficial for presentation attack detection is unevenly distributed in the image due to the various illumination and presentation attack instrument, so using randomly selected patches within the face region [1] fails to explore the full potential of the local representation for discriminative presentation attack detection. Considering this, Yang *et al.* [22] proposed to learn the offset of the discriminative regions based on extracted CNN features, which is highly dependent on the initial localization and thus is not enough to discover the salient local part for further presentation attack detection. Therefore, how to better discover the discriminative regions is of crucial importance for more accurate and robust face presentation attack detection.

Motivated by this, we propose a salience-aware face presentation attack detection (SAFPAD) approach, which employs deep reinforcement learning to locate the salient local parts and then exploits the most salient local part information to improve the face presentation attack detection performance. Additionally, we propose a salient local feature encoding

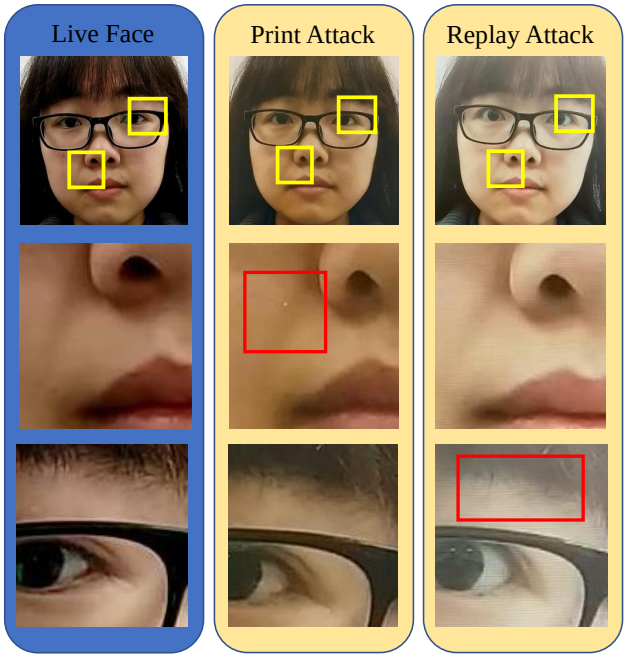

Fig. 1. Motivations of the proposed SAFPAD approach. In face presentation attack detection, it is crucial to exploit the discriminative cues of the image, which are unevenly distributed in an image and differently located for different attacks. The top row shows the selected location of two salient patch examples. The bottom two rows show the enlarged local patches. Moreover, left, middle, and right columns come from the bona fide face, print attack, and replay attack, respectively. The red boxes represent the locations of artifacts in this figure. The comparison between same region of bona fide face and print attack shows that the artifacts of color distortion appear around the patch near the nose and the hair. However, as for replay attack, artifacts of moire pattern appear around a different patch which is near the eyes. Therefore, it is meaningful to exploit the salient local part information in face images.

network, which focuses on the salient part to reduce the adverse effects of redundant information in the face image. Finally, we combine the high-level feature and the salient local feature to obtain the fused feature, which consists of both the contextual information and the salient local information. The core problem is that the policy for selecting the salient locations is non-differentiable, since the salience selection process lacks a direct supervision signal. To address this, we formulate the salient part selection process as a Markov decision process which is able to be effectively optimized by reinforcement learning algorithms, where we define each action to predict the salient patch location according to the current state. We design each state to incorporate two parts, the fused feature and the previous hidden state information, and the reward to indirectly adopt a hybrid loss to guide the agent. Furthermore, to solve the dilemma of exploration and exploitation for salient locations during the training stage, we jointly train the recurrent model and the multiscale feature fusion network using deep reinforcement learning supervised with a hybrid loss based on backpropagation. As a consequence, the optimization of the multiscale feature fusion network provides contextual information, which in turn assists the proposed SAFPAD model to explore more salient local patches. Our approach achieves very competitive performance on multiple public face presentation attack detection datasets.

II. Related Work

In this section, we briefly review two related topics: 1) face presentation attack detection and 2) deep reinforcement learning.

A. Face presentation attack detection

Existing face presentation attack detection methods can be primarily categorized into three classes: texture-based methods, temporal-based methods, and hardware-based methods [11], [13], [14]. Texture-based methods aim to exploit the texture patterns for face presentation attack detection. Conventional methods used hand-crafted features such as LBP [16], [18], [23], HoG [17], [24], SIFT [25], and SURF [26], and employ binary classifiers to analyze the essential differences between bona fide faces and attack presentations. Boulkenafet *et al.* [14] proposed to learn various color image representations to study the effectiveness of the intrinsic disparities in the different color spaces (RGB, HSV and YCbCr). Garcia [27] applied Moiré patterns analysis to distinguish the bona fide face and attack presentations. With the development of deep learning, there are an increasing number of texture-based methods based on the CNN model for face presentation attack detection. Li *et al.* [19] utilized the pre-trained VGG-face model to extract features and employed the principle component analysis (PCA) method to reduce the dimensionality of features for classification. Atoum *et al.* [1] proposed a two-stream CNN-based approach and applied the auxiliary depth supervision. Shao *et al.* [28] adopted a multi-adversarial discriminative domain generalization approach to learn a generalized feature. However, all these methods focus on the entire image or several fixed regions rather than the salient part of the input image, which neglect the salient local part information which is crucial for face presentation attack detection.

For the second category, the input to various feature extractors is always a video. Temporal-based methods aim to distinguish the bona fide face and attack presentations based on temporal cues extracted from multiple frames. For example, Bao *et al.* [13] proposed to extract optical flow as dynamic textures other than the common facial motion such as eye-blinking and lip motion. Agarwal *et al.* [10] first applied redundant discrete wavelet transformation to a video and then extracted block-wise Haralick texture features from multiple frames. Similar to the texture-based methods, there exist a great number of temporal-based methods based on deep learning. Zhao *et al.* [29] proposed a volume local binary counting method which is a local spatio-temporal descriptor for recognition and representation of dynamic information. To extract local features and dense temporal features, Xu *et al.* [30] designed a temporal architecture combining LSTM units with CNN. Because of the non-invasive property, rPPG has extensive application prospects in face presentation attack detection [31]–[33]. Liu *et al.* presented a solution [34] with the rPPG information extracted from local facial parts via CHROM [35] to combine the global signal and spatial information. Moreover, Liu *et al.* [36] proposed a CNN-RNN model to estimate both depth information and temporal

signals under auxiliary supervision for face presentation attack detection. However, all the temporal-based methods are based on videos, which demand more consumption of computing resources resulting in inconvenience for applications in real life.

The hardware-based methods are also worth considering because nowadays more and more laptops and phones are equipped with various sensors, such as thermal sensors and structured light sensors. This motivates researchers to deal with face presentation attack detection using hardwares. Raghavendra *et al.* [37] proposed a novel approach to explore the variance of the focuses between different depth images rendered via the light field camera that can reveal the attack presentations in turn. Further, Chan *et al.* [6] adopted flashes to reveal the differences between bona fide users and attack presentations and reduced the effect of environmental factors. For multi-modal data such as from color, near-infrared, depth, and thermal data, George *et al.* [38] adopted an approach based on the multi-channel Convolutional Neural Network. Moreover, Yu *et al.* [39] proposed a multi-modal version central difference convolutional networks to explore intrinsic spoof texture among three modalities (RGB, infrared, and depth). Although hardware-based methods have brought a lot of progress to face presentation attack detection, there is an inevitable problem that hardware-based methods require various sensors which will bring high costs in practical applications.

B. Deep Reinforcement Learning

Deep reinforcement learning has been shown to be effective for search-based problems [40]. Formally, unlike conventional learning-based optimization methods, reinforcement learning can be formulated as a Markov decision process (MDP) [41]. Deep learning recently assists reinforcement learning to deal with problems that were out of capacity in the past [42], [43], and combining reinforcement learning with deep learning methods begins to draw more and more attention. Deep reinforcement learning algorithms have achieved very promising results [40], [44], [45], which can be primarily divided into two categories: policy-gradient based and deep-Q-network based. For example, Silver *et al.* [45] proposed a program named AlphaGo based on Monte-Carlo tree search with deep reinforcement learning, which defeated the human world champion when playing the game of Go. Moreover, Mnih *et al.* [40] tried to play ATARI games with deep Q-networks.

Deep reinforcement learning techniques have been applied in multiple visual tasks [46]–[48]. For example, Mathe *et al.* [49] proposed a region proposal network to probe extensive exploration-accuracy trade-offs for object detection with deep reinforcement learning. Wang *et al.* [50] formulated the online key decision problem of dynamic video segmentation as a deep reinforcement learning process and learned an effective and efficient scheduling policy from expert annotations about decision history. For the visual tracking tasks, Yun *et al.* [51] designed action-decision networks to reduce computation complexity in tracking with deep reinforcement learning. Yang *et al.* [52] present the first inverse reinforcement learning model to learn the hidden reward function and policy adopted by humans during visual search. Shao *et al.* [53] modeled the 6D pose refinement problem as a Markov Decision Process to use only 2D image annotations for weakly-supervised 6D pose refinement. In the research field related to face, deep reinforcement learning also has a wide range of applications. For example, Cao *et al.* [54] employed an attention-aware framework to incorporate features among different facial parts for face hallucination with deep reinforcement learning. Moreover, Duong *et al.* [55] utilized the deep reinforcement learning method to guarantee consistency of the visual identity in synthesized faces. In addition, Wang *et al.* [56] proposed a ethnicity balance network based on reinforcement learning to learn better performance for different ethnicities face recognition based on large margin losses. However, there is still room for improvement in the use of deep reinforcement learning algorithms [48].

The combination of deep learning and reinforcement learning is capable of various tasks, such as ATARI games [40] and face hallucination [54]. In this paper, we develop a recurrent salience-aware model and employ deep reinforcement learning to exploit the salience in images for face presentation attack detection.

III. Proposed Approach

In this section, we first present the formulation of our SAFPAD model in detail. Then, we explain how the multiscale feature fusion architecture works. Finally, we introduce how to optimize the SAFPAD model and the implementation details. Fig. 2 shows the overall network architecture of our proposed approach.

A. Recurrent Salience-Aware Module

We employ the recurrent salience-aware model to investigate salient patches with the observation of multiscale information from the image. At the t-th step, according to the current state, the agent takes action to predict the location of the salient patch. Given the salient patch, the multiscale feature fusion network will obtain and fuse the high-level feature and the salient local feature. After that, the recurrent model updates the state by combining the current state and the fused feature. At the same time, the recurrent model utilizes the fused feature to predict the face image classification: bona fide face or attack presentation. The recurrent model iteratively predicts the most salient patch and classifies the fused feature until the maximum step is reached. At last, to guide the agent to predict the salient locations, we design a reward that is measured by the cross entropy of the classification result and the labels. To avoid all the selected patches gathering together to the most salient location, an auxiliary variance reward is applied. Next, we will introduce the formulation of state, action, and reward in detail.

State: To ensure that the agent can take optimal action with multiscale feature and contextual information, the state s_t at the t-th step consists of two parts: the fused feature and the previous state information. Given the input image x, we can extract the high-level feature $g_{img} = F_{img}(x)$, where $F_{img}(\cdot)$ denotes the high-level feature encoding network. Similarly, we can extract the salient local feature $g_{loc,t}$

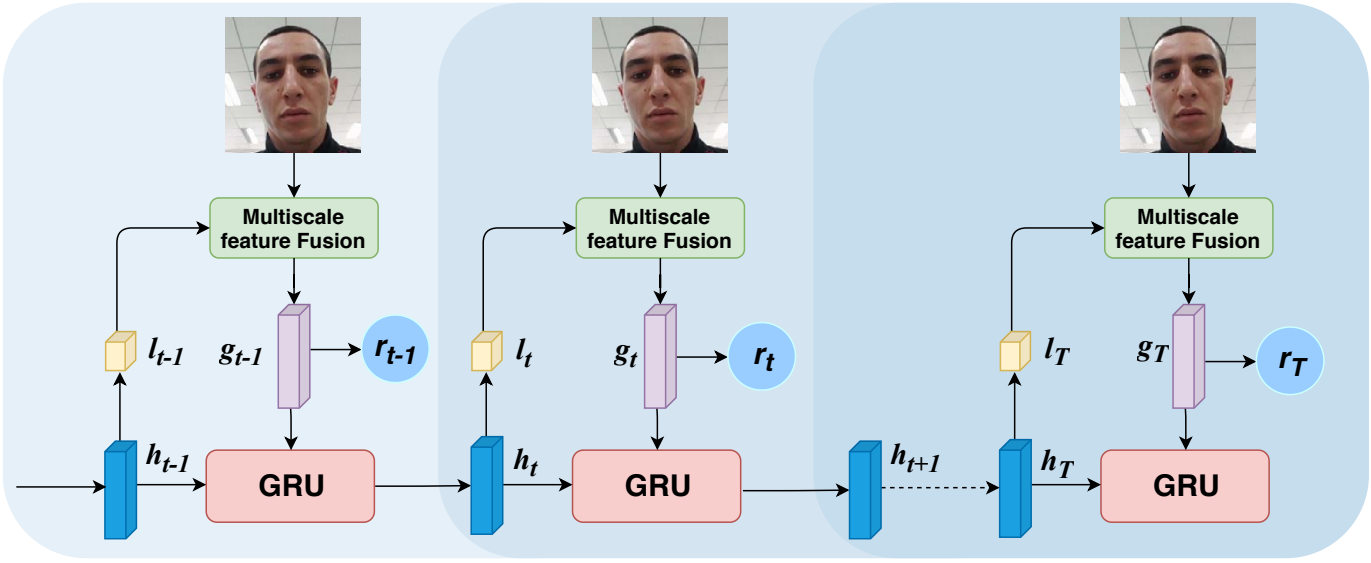

Fig. 2. The network architecture of the proposed recurrent salience-aware reinforcement learning framework. At the t-th step, according to the current state h_t, the agent takes action to predict the salient location l_t. Given the face image and the salient location l_t, we obtain the multiscale fused feature g_t. After that, the GRU updates the state by combining the current state h_{t-1} and the fused feature g_t. At the same time, we utilize a classification network to classify the image and calculate the reward r_t. Overall, the recurrent salience-aware model and the multiscale feature fusion architecture jointly update the network parameters with reinforcement learning.

with the input image x and the current salient location l_t: $g_{loc,t} = F_{loc}(x, l_t)$, where $F_{loc}(\cdot)$ denotes the salient local feature encoding network. After that, the fused feature g_t is formed by concatenating the high-level feature g_{img} and the salient local feature $g_{loc,t}$: $g_t = G(g_{img}, g_{loc,t})$. To remember and utilize the previous state information, we employ a GRU (Gated Recurrent Unit) [57] $F_h(\cdot)$ to generate the next state:

$$s_{t+1} = h_{t+1} = F_h(g_t, h_t), \tag{1}$$

where h_t is the hidden variable of the current state. This GRU can assist the agent to observe all the previous states and actions to predict salient location.

Action: At the t-th step, the agent predicts the salient patch location l_t according to the current state s_t. We formulate the salient location as a Gaussian distribution:

$$l_t \sim N(l_t|\mu, \Sigma). \tag{2}$$

When the $F_{img}(\cdot)$ extracts the high-level feature, the reduction in the feature map size will lead to the loss of location information. To estimate parameters of the Gaussian distribution, the agent utilizes a policy network to generate a probability distribution map of the salient location $\{l_t = (x, y)|0 \le x \le W, 0 \le y \le H\}$, where W, H is the size of the high-level feature map:

$$\mathrm{P}(l_t = (x, y)|h_t) \ = F_\pi(h_t; \theta_\pi), \tag{3}$$

where $F_\pi(\cdot)$ denotes the policy network and θ_π denotes the parameters. Then the agent selects the location with highest probability as the mean μ of the Gaussian distribution:

$$\mu = \arg\max_{(x,y)} \ \|\mathrm{P}(l_t = (x, y)|h_t)\|. \tag{4}$$

As for the variance of the Gaussian distribution, we set an appropriate value to avoid excessive overlap of different salient location distribution l_t. Given the mean μ and the variance Σ, the agent normalizes the salient location for selecting a patch in the image x.

Reward: We employ a hybrid reward to guide the agent to explore the most salient patches of the input image. In order to classify the bona fide face and attack presentations, we utilize a classification network $F_c(\cdot)$ to predict the result $C_t = F_c(g_t)$ and minimize the cross entropy. The classification network is able to observe both the salient local information and global image information. We assume the ground-truth label of the input image is L, then the reward at the t-th step can be set as:

$$r_t = \begin{cases} 1 & C_t = L, t = T \\ 0 & \text{otherwise.} \end{cases} \tag{5}$$

On the other hand, there is a tendency for the agent to focus on the most salient location all the time. We introduce a location reward r_l which adopts the variance of the location to overcome this dilemma: $r_l = \mathrm{Var}(l_t)$. Finally, we calculate the total reward R as follows:

$$R = \sum_{t=1}^{T} \gamma^{t-1} r_t + \lambda_l r_l, \tag{6}$$

where γ is the discount factor and the λ_l is the hyper-parameter.

B. Multiscale Feature Fusion

High-level Feature Encoding Network: We design the high-level feature encoding network to extract the global information of the input image. Fig. 3 shows the network architecture of the multiscale feature fusion. Inspired by [36], to extract more discriminative feature, the high-level feature encoding network consists of a convolutional layer, three

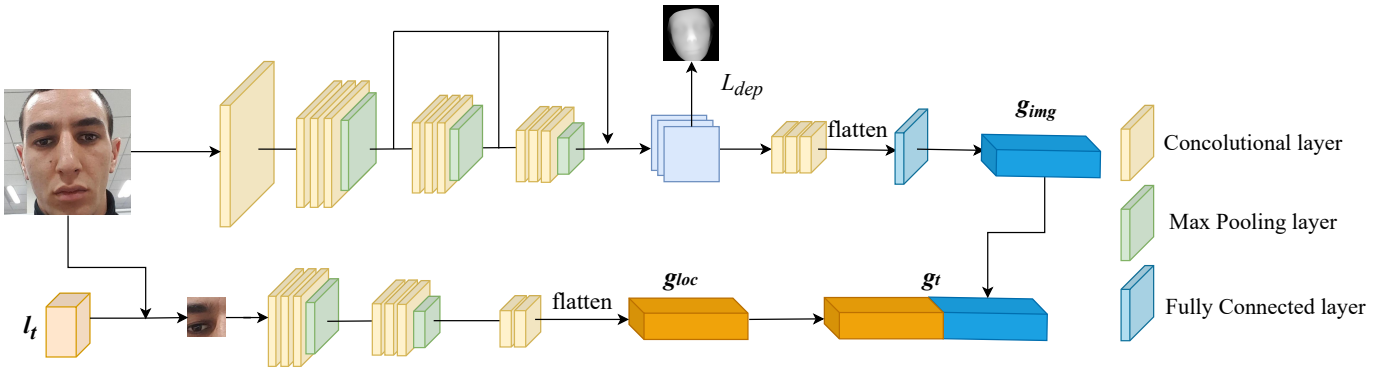

Fig. 3. The proposed multiscale feature fusion architecture. The kernel size of convolution layers is 3×3 with stride 1. All the max pooling layers have kernel size 3×3 and stride 2. The feature dimension of the fully connected layers is set as 256. The size of the feature maps concatenated with a short-cut connection from three blocks is 32×32.

blocks and three convolutional layers. Three convolutional layers and one pooling layer compose the block. Similar to the residual network [58], we resize the feature maps of three blocks to 32×32 and use a short-cut connection to concatenate them. Then the following three convolutional layers utilized the concatenated feature map for estimating the depth map. To obtain the ground-truth, we use the dense face alignment method PRNet [59] to estimate the depth map D of the bona fide face, and the depth map D of the attack presentation is set to a zero map. The depth map loss L_{dep} is applied to supervise the network:

$$L_{dep} = ||F_{dep}(x; \theta_{dep}) - D||_1^2, \tag{7}$$

where the $F_{dep}(\cdot)$ is the network to estimate the depth map and θ_{dep} denotes the parameters. After the convolutional layers, we adopt a flatten layer and a fully connected layer to generate the high-level feature g_{img}.

Salient Local Feature Encoding Network: After obtaining the contextual and discriminative global information, we employ a network to extract the local information of the salient part in the image. Given the input image x and the salient location l_t, we crop a fixed-size patch from x at location l_t as the salient part. Then the salient part is fed into the following two blocks and two additional convolutional layers. Each block includes three convolutional layers and one pooling layer. After that, the feature map is flattened to a feature vector g_{loc}. The salient local feature g_{loc} can provide information of the patch at the t-th location l_t.

Finally, we directly concatenate the high-level feature g_{img} and the salient local feature $g_{loc,t}$ to get the fused feature g_t. The fused feature g_t consists of salient local information and the contextual information to guide both the agent and classifier.

C. Deep Reinforcement Learning for SAFPAD Model

The policy for selecting the salient locations in the SAFPAD model is undifferentiable. In addition, to avoid the optimization of the SAFPAD model getting stuck in local optimal solution due to initial salient location during the training stage, we propose the deep reinforcement learning method to jointly train the recurrent salience-aware model and the multiscale feature fusion network.

The policy of the agent and the state transition probability compose the decision process probability distribution $P(\tau|\theta)$, then our goal is to maximize the expected reward under this distribution:

$$J(\theta) = \mathbb{E}_{P(\tau|\theta)}[\sum_{t=1}^{T} \gamma^{t-1} r_t + \lambda_l r_l] = \mathbb{E}_{P(\tau|\theta)}[R], \tag{8}$$

Generally, it is difficult to estimate the decision process probability distribution $P(\tau|\theta)$. Following [60], we adopt a sample approximation of the gradient with the Monte Carlo method:

$$\begin{aligned} \nabla_\theta J &= \mathbb{E}_{P(\tau|\theta)}[\nabla_\theta log P(\tau|\theta) R] \\ &= \sum_{t=1}^{T} \mathbb{E}_{P(\tau|\theta)}[\nabla_\theta log \pi(l_t|s_t) R] \\ &\approx \frac{1}{M} \sum_{m=1}^{M} \sum_{t=1}^{T} \nabla_\theta log \pi(l_{t,m}|s_{t,m}) R_m, \end{aligned} \tag{9}$$

where $\pi(\cdot)$ denotes the policy network and $m = 1, \ldots, M$ denotes the m-th episodes in the Monte Carlo method. The above sample approximation of the gradient is unbiased, but it may lead to high variance as a result of the moving rewards. For this reason, we utilize a sample approximation to the gradient with baseline:

$$\nabla_\theta J \approx \frac{1}{M} \sum_{m=1}^{M} \sum_{t=1}^{T} \nabla_\theta log \pi(l_{t,m}|s_{t,m})(R_m - b), \tag{10}$$

where b is the baseline related to the state but not to the action. In terms of value, the two gradients are equal, but the second gradient reduces the variance of sample approximation. To estimate the baseline, we design a value network to minimize the MSE (Mean Squared Error) between the reward and the baseline.

On the other hand, we employ a cross entropy loss L_{cls} to optimize the classification network:

$$L_{cls} = -\frac{1}{N} \sum_{n=1}^{N} (y_n log(\hat{y}_n) + (1 - y_n) log(1 - \hat{y}_n)), \tag{11}$$

Algorithm 1 : Optimization of SAFPAD model

Input: image x, the fixed size of salient patches $W \times H$, and the number of salient patches T.

Output: Weights of SAFPAD model θ_π, θ_h, high-level feature encoding network θ_{img}, and salient local feature encoding network θ_{loc}.

Initialize the weights θ_π, θ_h, θ_{img}, and the θ_{loc};
for $b \leftarrow 1, 2 \ldots, Batchsize$ **do**
  for $t \leftarrow 0, 1 \ldots, T$ **do**
    if $t = 0$ **then**
      Initialize the l_0 and h_0 with 0;
      Obtain the fused feature g_0 with x and l_0;
    else
      Compute next state h_t with g_{t-1} and h_{t-1};
      Compute the salient location l_t with h_t;
      Obtain the fused feature g_t with x and l_t;
      Compute the reward r_t with g_t;
    end if
  end for
  Compute the hybrid reward as (6);
  Compute the depth map loss as (7);
  Compute the RL gradients as (10);
  Compute the cross entropy loss as (11);
  Update the weights θ_π, θ_h, θ_{img}, and θ_{loc}.
end for

return The weights θ_π, θ_h, θ_{img}, and θ_{loc}.

where y_n is the ground truth label of the n-th input image and $\hat{y}_n$ denotes the predicted probability of the classification network. This cross entropy loss assists the SAFPAD model to calculate reward.

The recurrent salience-aware model is optimized with a hybrid reward based on REINFORCE algorithm and the multiscale feature fusion network is supervised with a hybrid loss based on back-propagation. The recurrent salience-aware reinforcement learning framework jointly updates the network parameters to explore the most salient local parts in the input image.

We summarize the details of the proposed recurrent salience-aware reinforcement learning framework in Algorithm 1.

IV. Experiments

We evaluated the proposed model on five face presentation attack detection datasets, including print, video replay, and mask attacks: CASIA-MFSD [61], Replay-Attack [62], Oulu-NPU [63], SiW [36], and HQ-WMCA [64]. The detailed summary of all datasets is presented in Table I, which shows some statistics of the face presentation attack detection datasets. The CASIA-MFSD and Replay-Attack datasets are used for the cross-dataset testing, and the Oulu-NPU, SiW, and HQ-WMCA datasets are selected for the intra-dataset testing.

A. Datasets and Metrics

CASIA-MFSD [61]: This dataset consists of 50 subjects, and each subject contains 12 videos (3 bona fide and 9 attack). The overall number of videos in the dataset is 600. There are three kinds of imaging qualities composing of high, normal and low qualities and three kinds of attacks. For each subject, the attack presentations are taken under three quality conditions. As shown in Fig. 4, the attack presentations consist of three kinds of attack presentations, including warped print attack, print attack with eye region paper cut and replay attack in three qualities, respectively.

Fig. 4. Example images of bona fide face and attack presentations from the CASIA-MFSD dataset. On the left, the sample is a bona fide face. The other samples from left to right show examples of original attack presentations, attack presentations generated by Google Nexus 5 and attack presentations generated by iPhone 6.

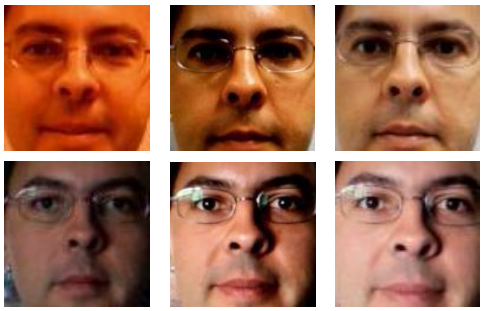

Fig. 5. Example images of bona fide faces and attack presentations from the Replay Attack dataset. The top row corresponds to the images taken from the controlled condition and the bottom row presents the images from the adverse condition. From the left to the right: bona fide faces, the high definition replay and print attacks. Examples of print and replay attack faces are taken with the front camera of Sony XPERIA C5 Ultra Dual.

Replay-Attack [62]: This dataset contains 200 bona fide videos and 1000 attack videos from 50 subjects. Further, for each subject, two extra bona fide videos are captured as enrollment data. Thus, the overall number of videos in the dataset is 1300. For each subject, bona fide face video and four attack video sequences are recorded with adverse and controlled background scenes. As for attack samples, three attack presentation types are adopted, i.e., print attack, digital photo attack, and replay attack. As shown in Fig. 5, the attack videos are taken in front of handheld or fixed support mediums.

Oulu-NPU [63]: This dataset includes 990 bona fide videos and 3,960 attack videos from 55 subjects. These videos sequences were taken using the front cameras of six display devices (Samsung Galaxy S6 edge, MEIZU X5, HTC Desire EYE, Sony XPERIA C5 Ultra Dual, ASUS Zenfone Selfie and OPPO N3) in three different sessions with various background scenes and illumination conditions. As shown in Fig. 6, the attacks were implemented using two printers and two display devices (Dekk 1905FP and Macbook Retina). There are four Protocols about Oulu-NPU for the evaluation of

TABLE I
THE SUMMARY OF PUBLIC-DOMAIN DATASETS IN THE EXPERIMENTS.

Datasets	CASIA-MFSD	Replay Attack	Oulu-NPU	SiW	HQ-WMCA
Year	2012	2012	2017	2018	2020
# Subs.	50	50	55	165	51
# Sess.	3	1	3	4	3
bona fide/attack	150/450	200/1000	1980/3960	1320/3300	555/2349
pose range	Frontal	Frontal	Frontal	$[-90°, 90°]$	Frontal
dif. expres.	No	No	No	Yes	No
Extra light	No	Yes	Yes	Yes	Yes
Attack Presentations Print/Replay	1/1	1/2	2/2	2/4	4/3
Resolution	320×240	1280×720	1920×1080	1920×1080	1920×1200
Acquisition devices	Sony NEX5 webcam	Macbook Cannon SX 150IS	Galaxy S6 HTC MEIZU X5 ASUS Sony C5 OPPO N3	-	Basler Xenics Intel Realsense
Display devices	iPad	iPhone 3GS iPad	Dell 1905FP Macbook Retina	iPad Pro iPhone 7 Galaxy S8 Asus	CX c224e Epson-XP
Subs. ethnicity	Asian 100%	Caucasian 76% Asian 22% African 2%	-	Caucasion 35% Asian 35% African 7% Indian 23%	-

Fig. 6. Example images of bona fide face and attack presentations from the Oulu-NPU dataset. In the left column, we show the example of a bona fide face. In the other columns from left to right, the examples of attack presentations are from printer1, printer2, display device1 and display device2.

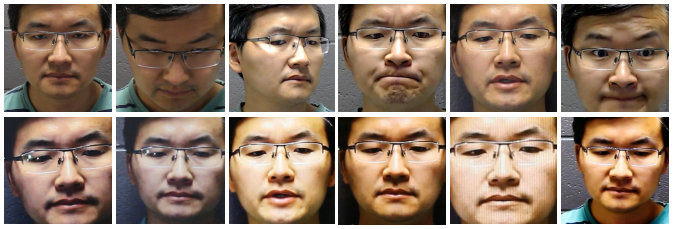

Fig. 7. Example images of bona fide faces and attack presentations from the SiW dataset. The dataset provides two print attacks and four display attacks for each subject. Sample images of bona fide faces with different gestures and expressions are shown in the first row. The second row shows the sample images of attack presentations taken by the printers and display cameras consisting of 6 presentation attack instruments: printer1, printer2, Samsung S8, iPhone, iPad, and PC screen, respectively.

the generalization capability of the face presentation attack detection algorithms. Protocol 1 evaluates on the illumination and background scene variation, Protocol 2 examines the impact of different types of presentation attack instrument, Protocol 3 studies the effect of camera devices variation, and

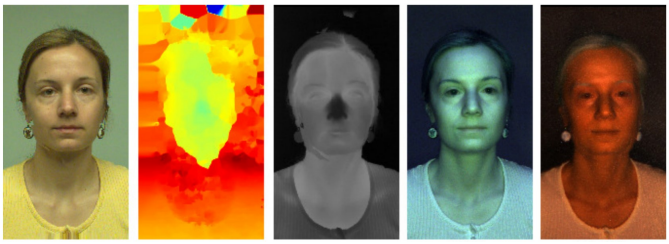

Fig. 8. Example images of bona fide faces and attack presentations from the HQ-WMCA dataset. From left to right, the images respectively come from color, depth, thermal, NIR, and SWIR channels. This dataset combines a subset of images from the corresponding spectra to obtain NIR composite and SWIR composite.

Protocol 4 considers all the factors above, which is the most challenging Protocol.

SiW [36]: This dataset includes 1320 bona fide videos and 3300 attack videos from 165 subjects, and the dataset considers variations in illuminations, poses, expressions, as shown in Fig. 7. There are four sessions for the collected bona fide videos. In Session 1, the subject keeps his head away from the camera with varying distances. In Session 2, the subject chooses different yaw angles of the head within $[-90°, 90°]$, and makes various facial expressions. In Sessions 3 and 4, the subject still follows the Sessions 1,2, but the collector changes the point light source shining on the face from different directions. Three Protocols from SiW deal with respective variations in facial expression and pose, cross presentation attack instrument and cross PA.

HQ-WMCA [64]: This dataset consists of 1233 bona fide videos and 1826 presentation attack videos from 51 different

TABLE II
THE PARTITIONING DETAILS OF PUBLIC-DOMAIN DATASETS IN THE EXPERIMENTS.

Protocol	subset	Bona fide videos	Attack videos	Total videos
OULU 1	Train	240	960	1200
	Dev	180	720	900
	Test	120	480	600
OULU 2	Train	360	720	1080
	Dev	270	540	810
	Test	360	720	1080
OULU 3	Train	300	1200	1500
	Dev	225	900	1125
	Test	60	240	300
OULU 4	Train	200	400	600
	Dev	150	300	450
	Test	20	40	60
SiW	Train	720	1800	2400
	Dev	-	-	-
	Test	600	1500	2100
Casia	Train	60	180	240
	Dev	-	-	-
	Test	90	270	360
Replay	Train	60	300	360
	Dev	60	300	360
	Test	80	400	480
HQ-WMCA	Train	228	742	970
	Dev	823	300	1123
	Test	182	784	966

subjects. All the videos from this dataset are multi-channel sequences. Different from the previous dataset, this dataset includes a wide variety of attacks types such as obfuscation attacks and impersonation ones. Specifically, all the attacks consist of glasses, mannequin, print, replay, makeup, tattoo, tattoo, wigs and many masks (rigid mask, paper mask, and flexible mask). The database consists of several channels including color, depth, thermal, near infra-red, and short-wave infrared, as shown in Fig. 8.

We employed Attack Presentation Classification Error Rate (APCER) [65], Bona Fide Presentation Classification Error Rate (BPCER) [65] and ACER [66] as the evaluation metrics:

$$ACER = \frac{APCER + BPCER}{2} \tag{12}$$

B. Implementation Details

We provide partitioning details for all the intra-dataset protocols in Table II in order to see the number of training/validation/testing samples. We implemented our approach with PyTorch framework. As shown in Table III and IV, we list the kernel size, stride and output size of each layer. The detailed network structure of the high-level feature encoding network and the salient local feature encoding network is listed. The human vision system generally attends to multiple salient objects one by one. This inspires us to employ the GRU as the basic framework to sequentially capture salient local clues. Considering the size of both input images and salient patches, we set the maximum number of steps $T = 8$ which was also the number of salient local patches. In the multiscale feature fusion network, each convolutional layer was followed by a rectified linear unit (ReLU) [67] and batch normalization (BN) [68] layer. The dimension of hidden layers in the GRU was set as 512. The samples in all the five face presentation attack detection datasets are videos, and we

TABLE III
THE DETAILED NETWORK STRUCTURE OF HIGH-LEVEL FEATURE ENCODING NETWORK.

Layer	kernel	stride	output size
input			$256 \times 256 \times 3$
conv0	3×3	1	$256 \times 256 \times 64$
conv1	3×3	1	$256 \times 256 \times 128$
conv2	3×3	1	$256 \times 256 \times 196$
conv3	3×3	1	$256 \times 256 \times 128$
pool1	-	2	$128 \times 128 \times 128$
conv4	3×3	1	$128 \times 128 \times 128$
conv5	3×3	1	$128 \times 128 \times 196$
conv6	3×3	1	$128 \times 128 \times 128$
pool2	-	2	$64 \times 64 \times 128$
conv7	3×3	1	$64 \times 64 \times 128$
conv8	3×3	1	$64 \times 64 \times 196$
conv9	3×3	1	$64 \times 64 \times 128$
pool3	-	2	$32 \times 32 \times 128$
short cut connection pool1+pool2+pool3			
conv10	3×3	1	$32 \times 32 \times 128$
conv11	3×3	1	$32 \times 32 \times 64$
conv12	3×3	1	$32 \times 32 \times 1$

TABLE IV
THE DETAILED NETWORK STRUCTURE OF SALIENT LOCAL FEATURE ENCODING NETWORK.

Layer	kernel	stride	output size
input			$64 \times 64 \times 3$
conv0	3×3	1	$64 \times 64 \times 32$
conv1	3×3	1	$64 \times 64 \times 48$
conv2	3×3	1	$64 \times 64 \times 32$
pool1	-	2	$32 \times 32 \times 32$
conv3	3×3	1	$32 \times 32 \times 32$
conv4	3×3	1	$32 \times 32 \times 48$
conv5	3×3	1	$32 \times 32 \times 32$
pool2	-	2	$16 \times 16 \times 32$
conv6	3×3	1	$16 \times 16 \times 16$
conv7	3×3	1	$16 \times 16 \times 1$
flatten	-	-	256×1

randomly extracted 10 frames from the video. After detecting the face images from these 10 random frames, we resized all images to the fixed size 256×256 and extracted salient patches from the resized images. Moreover, the salient patch size $W \times H$ was set as 64×64. We employed the SAFPAD reinforcement learning framework to respectively provide 10 classification scores and obtained final prediction according to the majority voting principle. The network parameters were updated using Adam [69] optimizer with the batch size of 16 and the learning rate of 0.0003.

C. Results of Intra-dataset Testing

The intra-dataset testing is conducted on the Oulu-NPU dataset and SiW dataset. For both datasets, we followed all the testing Protocols for the evaluation. The intra-dataset testing results including APCER, BPCER, and ACER are shown in Table V and Table VI. For each metric, the smaller the value is, the better performance the approach achieves.

TABLE V
COMPARISONS OF THE RESULTS OF INTRA-DATASET TESTING ON ALL PROTOCOLS OF THE OULU-NPU DATASET.

Prot.	Method	APCER (%)	BPCER (%)	ACER (%)
1	CPqD [11]	2.9	10.8	6.9
	MILHP [70]	8.3	**0.8**	4.6
	GRADIANT [11]	1.3	12.5	6.9
	Motion blur [71]	7.7	10.5	9.1
	Soft-attention [72]	5.1	6.7	5.9
	Auxiliary [36]	1.6	1.6	1.6
	FaceDs [12]	1.2	1.7	1.5
	STASN [22]	1.2	2.5	1.9
	SAFPAD	**0.6**	1.2	**0.9**
2	MixedFASNet [11]	9.7	2.5	6.1
	MILHP [70]	8.3	0.8	4.6
	GRADIANT [11]	3.1	1.9	2.5
	Motion blur [71]	5.4	5.7	5.6
	Soft-attention [72]	7.6	2.2	4.9
	Auxiliary [36]	2.7	2.7	2.7
	FaceDs [12]	4.2	4.4	4.3
	STASN [22]	4.2	**0.3**	**2.2**
	SAFPAD	**2.2**	2.6	2.4
3	MixedFASNet [11]	5.3 ± 6.7	7.8 ± 5.5	6.5 ± 4.6
	MILHP [70]	$\mathbf{1.5 \pm 1.2}$	6.4 ± 6.6	4.0 ± 2.9
	GRADIANT [11]	2.6 ± 3.9	5.0 ± 5.3	3.8 ± 2.4
	Motion blur [71]	5.1 ± 4.9	10.4 ± 10.1	7.8 ± 6.5
	Soft-attention [72]	3.9 ± 2.8	7.3 ± 1.1	5.6 ± 1.6
	Auxiliary [36]	2.7 ± 1.3	3.1 ± 1.7	2.9 ± 1.5
	FaceDs [12]	4.0 ± 1.8	3.8 ± 1.2	3.6 ± 1.6
	STASN [22]	4.7 ± 3.9	$\mathbf{0.9 \pm 1.2}$	2.8 ± 1.6
	SAFPAD	2.2 ± 2.0	2.8 ± 3.7	$\mathbf{2.5 \pm 2.6}$
4	Massy_HNU [11]	35.8 ± 35.3	8.3 ± 4.1	22.1 ± 17.6
	MILHP [70]	15.8 ± 12.8	8.3 ± 15.7	12.0 ± 6.2
	GRADIANT [11]	5.0 ± 4.5	15.0 ± 7.1	10.0 ± 5.0
	Motion blur [71]	$\mathbf{4.9 \pm 2.4}$	7.9 ± 6.7	6.4 ± 5.3
	Soft-attention [72]	11.3 ± 3.9	9.7 ± 4.8	9.8 ± 4.2
	Auxiliary [36]	9.3 ± 5.6	10.4 ± 6.0	9.5 ± 6.0
	FaceDs [12]	5.1 ± 6.3	6.1 ± 5.1	$\mathbf{5.6 \pm 5.7}$
	STASN [22]	6.7 ± 10.6	8.3 ± 8.4	7.5 ± 4.7
	SAFPAD	6.6 ± 7.1	$\mathbf{6.0 \pm 5.5}$	6.3 ± 5.9

TABLE VI
COMPARISONS OF THE RESULTS OF INTRA-DATASET TESTING ON THREE PROTOCOLS OF THE SIW DATASET.

Prot.	Method	APCER (%)	BPCER (%)	ACER (%)
1	Auxiliary	3.58	3.58	3.58
	STASN	1.00	1.00	1.00
	SAFPAD	**0.89**	**0.97**	**0.93**
2	Auxiliary	0.57 ± 0.69	0.57 ± 0.69	0.57 ± 0.69
	STASN	$\mathbf{0.28 \pm 0.05}$	$\mathbf{0.28 \pm 0.05}$	$\mathbf{0.28 \pm 0.05}$
	SAFPAD	0.39 ± 0.31	0.29 ± 0.23	0.34 ± 0.27
3	Auxiliary	8.31 ± 3.80	8.31 ± 3.80	8.31 ± 3.80
	STASN	12.10 ± 1.50	12.10 ± 1.50	12.10 ± 1.50
	SAFPAD	$\mathbf{8.26 \pm 1.98}$	$\mathbf{8.12 \pm 2.17}$	$\mathbf{8.19 \pm 2.24}$

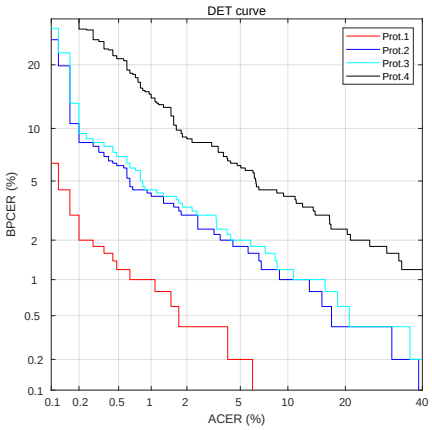

Fig. 9. The DET curves for all the four protocols on the OULU-NPU dataset.

We compared proposed SAFPAD model with a wide variety of methods, including CPqD [11], GRADIANT [11], MILHP [70], MixedFASNet [11], MassyHNU [11], Motion blur [71], Soft-attention [72], Auxiliary [36], FaceDs [12] and STASN [22]. Especially for STASN, this work provides two approaches and the second one utilizes private additional data. Thus it is not shown in Table V and Table VI for the sake of fairness. The best results of these above methods are shown in bold, and the second best performance is underlined.

For the Oulu-NPU dataset, our proposed approach achieved state-of-the-art performance on Protocol 1 and Protocol 3, and the ACER are respectively 0.9% and 2.5%. We had the second best ACER on Protocol 2 and Protocol 4. Still, SAFPAD achieved state-of-the-art APCER 2.2% on Protocol 2 and BPCER 6.0% on Protocol 4. This shows that the SAFPAD reinforcement learning framework is extremely effective in distinguishing the bona fide and attack samples. Our proposed approach achieved the second best ACER on Protocol 4 for OULU-NPU which is the most challenging Protocol. The slightly worse performance may be due to the difference in the illuminations of different devices in the dataset.

On the SiW dataset, we achieved state-of-the-art performance on Protocol 1 and Protocol 3. Hence, compared with state-of-the-art methods, our method achieved very competitive performance. This suggests that the SAFPAD reinforcement learning framework is extremely effective in distinguishing the bona fide faces and attack presentations. Our proposed approach achieved the second best ACER on Protocol 2 for SiW which are the challenging Protocols. The slightly worse performance may be due to the difference in the resolution of different devices in the dataset. Therefore, our approach improves the generalization capability in response to complex practical face presentation attack detection environmental conditions.

In addition to selecting a single threshold, we need to plot DET curves [73], which visualize the detection performance for all possible thresholds. We present the results on OULU-NPU dataset and SiW dataset in Fig. 9 and Fig. 10. Then, it is possible to obtain the APCER/BPCER/ACER results with a threshold computed for the best ACER in the dev set. Thus we find and adopt the lowest cost point.

To evaluate our approach on additional PAI species, we conducted an experimental evaluation on the HQ-WMCA dataset with more sophisticated attacks such as masks or make-up. We followed the leave-one-out attack protocols: we left one attack out in the train and development set, at the same time, the test set consisted of bona fide samples and the attack presentations which was left out before. We present the ACER (%) results in Table VII and reported the values acquired with a threshold which is computed for BPCER 1% in development set. From this table, we can observe that, in addition to the common print and display attacks, our method can also achieve

TABLE VII
THE ACER (%) RESULTS ON THE LEAVE-ONE-OUT PROTOCOL OF HQ-WMCA DATASET.

Method	Flexiblemask	Glasses	Makeup	Mannequin	Papermask	Rigidmask	Tattoo	Replay	Mean±Std
PixBiS [74]	29.9	49.9	29.4	0.1	0.0	32.5	5.7	9.6	19.6±17.1
MCCNN [75]	14.2	32.7	22.0	1.5	7.1	33.7	4.2	36.6	19.0±13.2
ResDLAS [76]	23.5	50.0	33.8	1.0	2.6	31.0	5.7	15.5	20.3±16.2
SAFPAD	21.3	36.6	24.7	0.3	0.2	9.6	3.3	5.4	**12.7±13.4**

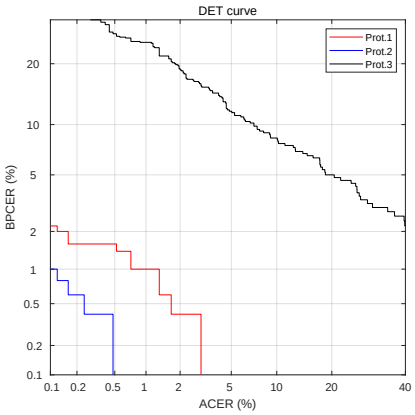

Fig. 10. The DET curves for all the three protocols on the SiW dataset.

TABLE VIII
COMPARISONS OF THE CROSS-DATASET TESTING RESULTS BETWEEN THE CASIA-MFSD DATASET AND REPLAY-ATTACK DATASET.

Method	Train	Test	Train	Test
	CASIA-MFSD	Replay Attack	Replay Attack	CASIA-MFSD
Motion [23]	50.2%		47.9%	
LBP-TOP [23]	49.7%		60.6%	
Motion-Mag [77]	50.1%		47.0%	
Spectral cubes [78]	34.4%		50.0%	
CNN [79]	48.5%		45.5%	
LBP [11]	47.0%		39.6%	
Colour Texture [14]	30.3%		37.7%	
Auxiliary [36]	<u>27.6%</u>		**28.4%**	
FaceDs [12]	28.5%		41.1%	
STASN [22]	31.5%		31.9%	
SAFPAD	**22.2%**		<u>28.9%</u>	

better performance when dealing with masks or make-up. Overall, our approach shows very competitive performance even compared with the multi-channel-based methods.

D. Results of Cross-dataset Testing

In face presentation attack detection, it is quite normal to conduct cross-dataset testing to evaluate the generalization capability of the proposed approaches. We adopted the CASIA-MFSD dataset and the Replay-Attack dataset to perform cross-dataset testing. First, we trained the model on CASIA-MFSD and test on Replay-Attack. Then we repeated the experiment after swapping the training dataset and the testing dataset. The ACER of various methods is shown in Table VIII.

Our proposed approach achieved state-of-the-art performance when testing on the Replay-Attack dataset, and the ACER is 22.2%. Our proposed method reduced the cross-testing errors on the Replay-Attack dataset by 5.4% relative to the previous SOTA. We had slightly worse but second best ACER when testing on the CASIA-MFSD dataset. To some extent, the SAFPAD model extracts the salient local information which depends on the image resolution. However, the image resolution of videos on the CASIA-MFSD dataset is higher than the Replay-Attack dataset. Overall, compared with state-of-the-art methods, our method achieved very competitive performance. This demonstrates that the SAFPAD reinforcement learning framework does extremely well in learning discriminative and generalizable cues. Therefore, our approach improves the generalization capability for real-world face presentation attack detection applications.

E. Parameter Analysis

In this subsection, we investigated the sensitivity and influences of the major parameters. The parameters analysis experiments were conducted on the Oulu-NPU Protocol 3. Additionally, we mainly analyzed the number of salient local patches (the time steps) and the size of the salient local patches.

Salient local patches number: We first explored how the ACER performance of the proposed approach changes when we chose different numbers of salient local patches (the time steps) in the SAFPAD model. Fig. 11 shows the ACER performance when the number of the salient local patches is different. The "T" indicates the number of GRU cell units in our recurrent model and denotes the number of salient local patches. The results demonstrate that when the number of salient local patches increases, the performance is getting better at the same time, since multiple salient local patches are able to extract more discriminative information for face presentation attack detection. Moreover, there is a gap in the performance between small salient local patches number and a big one, since a few salient local patches fail to locate the salient part and extract enough local information.

Salient local patches size: Further, we investigated the size of the salient local patches which is another important parameter of our SAFPAD model. To analyze the salient local patch size, we set the patches all square. Fig. 12 shows the performance when the salient local patches size is different. The "W(H)" indicates the salient local patches size in our SAFPAD model. The results show that when the size of salient local patches increases, the performance is getting better at the same time, since bigger salient local patches are able to extract more discriminative information for face presentation attack detection. We can observe that the mean ACER keeps decreasing and finally converges to a stable value. This result is reasonable, because as the parameters increase, the sizes of

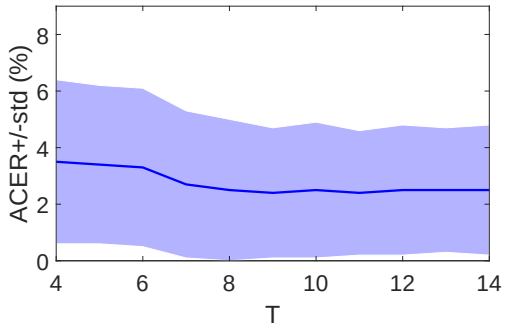

Fig. 11. The ACER(%) performance on the OULU-NPU Protocol 3 with different salient local patches number T. The blue line presents the mean ACER(%) value and the purple area presents the range of standard deviation.

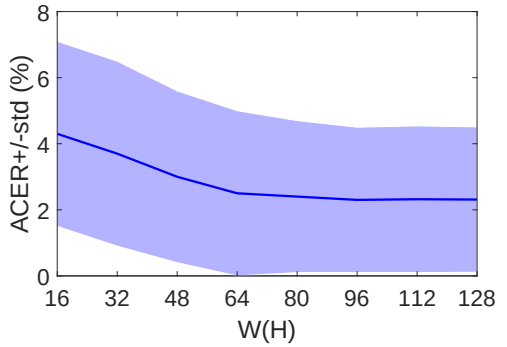

Fig. 12. The ACER(%) performance on the OULU-NPU Protocol 3 with different salient local patch size W(H). The blue line presents the mean ACER(%) value and the purple area presents the range of standard deviation.

TABLE IX
THE ABLATION STUDY RESULTS ON THE OULU-NPU DATASET IN TERMS OF PROTOCOL 1.

Prot.	Method	APCER (%)	BPCER (%)	ACER (%)
1	random location	1.9	3.1	2.5
	w/o high-level feature	5.9	7.5	6.7
	w/o RL	2.6	3.5	3.1
	w/o RSA module	2.9	3.7	3.3
	SAFPAD	**0.6**	**1.2**	**0.9**

salient patches are getting bigger, so the information obtained from the image will eventually reach the upper bound.

F. Ablation Study

We conducted four ablation studies on the Oulu-NPU dataset regarding the Protocol 1 to investigate the effects of various individual components in the SAFPAD reinforcement learning framework. The ablation study consisted of five settings which respectively adopt different network architectures. First, we replaced the salience-aware agent with a policy network which selects random locations of the input image. Second, we removed the high-level feature encoding network in the multiscale feature fusion and utilize salient local feature only. Third, we substituted the end-to-end backpropagation approach for the reinforcement learning framework. To further study the role of recurrent salience-aware (RSA) module, we completely discarded RSA module. Finally, the last model was just the proposed architecture. The results including APCER, BPCER, and ACER are shown in Table IX.

Effectiveness of the recurrent salience-aware module: After replacing the salience-aware locations with random locations or discarding RSA module, the proposed model has a poor performance which indicates that random locations mislead the classification. From the perspective of mathematical expectations, random locations treat the importance of local features in the image equally. Still, the spoofing cues in the image are unable to be evenly distributed in general. Therefore, the salience-aware model is vital to explore the most discriminative patches in the image to distinguish bona fide faces versus attack presentations. Moreover, it is more efficient to extract salient local information when given the high-level feature.

Effectiveness of the multiscale feature fusion: When we removed the high-level feature encoding network in the multiscale feature fusion, the model achieved substantially worst performance. The multiscale feature fusion is of great importance in our proposed approach, especially the high-level feature encoding network. The high-level feature plays two roles in the model, the first is providing the contextual information to guide the agent to generate the salient location and the second is extracting the global information of the image to assist the classifier to predict correct results. Under the supervision of the auxiliary depth map, the high-level feature encoding network is able to improve the generalization. Our approach improves the generalization capability for the complicated practical face presentation attack detection environmental conditions.

Effectiveness of the reinforcement learning: To substitute the end-to-end backpropagation approach for the reinforcement learning framework, we employed an agent network architecture similar to [22]. The new agent learned the offset of the patch instead of the location for the convenience of computing the gradients and end-to-end backpropagation. From the result, we can observe that the reinforcement learning framework is superior to the end-to-end backpropagation approach. Learning offset of the regions directly is not enough for exploring the salient local part due to the dependence on initial localization, yet the reinforcement learning encourages to exploit and explore the salient part of the input image.

G. Visualization and Analysis

The intra-dataset testing and the cross-dataset testing results demonstrate that our approach achieves very competitive performance. Further, the ablation study reveals that one of the core parts of our approach is the salience-aware location network. The recurrent salience-aware reinforcement learning framework jointly updates the network parameters to explore the most salient local parts in the input image. It is significant to study how the SAFPAD reinforcement learning framework works. We visualize the input images, salient location and resized local patches for videos on the Oulu-NPU dataset in Fig. 13. We show the successful presentation attack detection examples from the Oulu-NPU and SiW datasets which are correctly classified. The yellow boxes are the salient patches generated by the SAFPAD model.

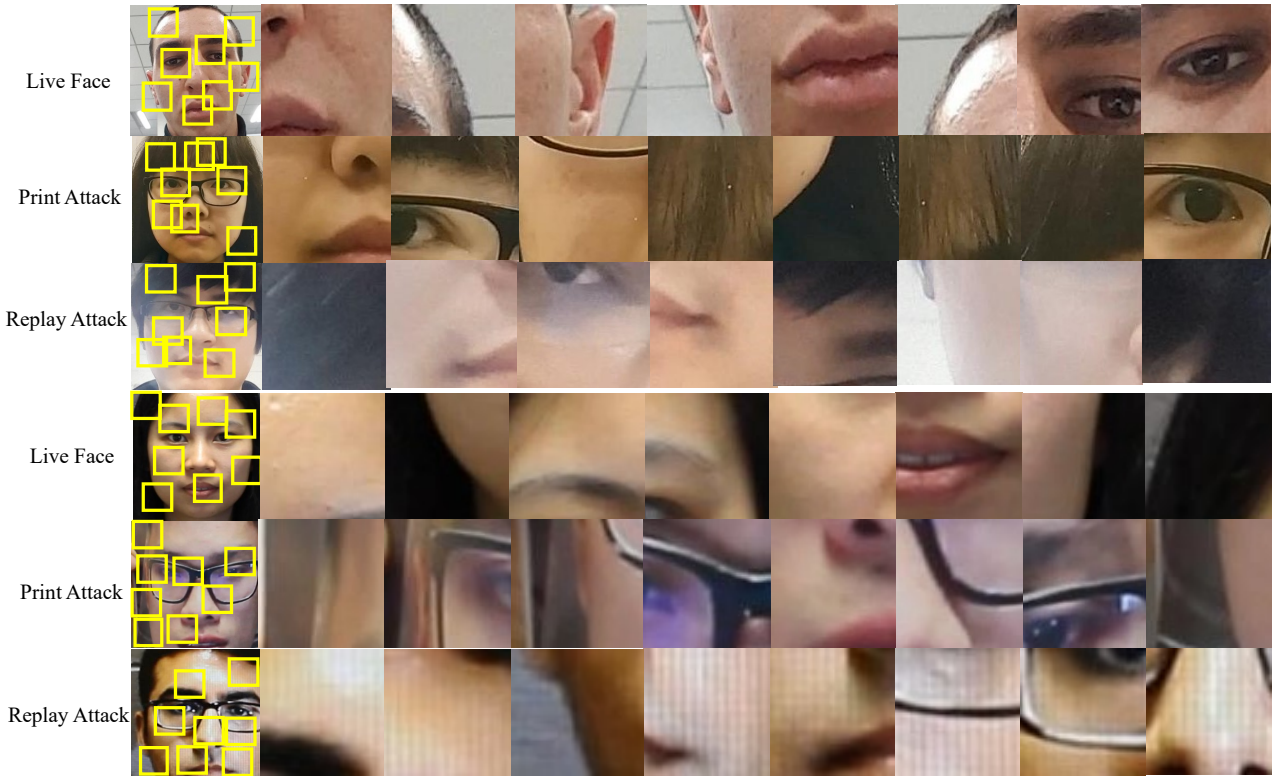

Fig. 13. The visualization of input images, salient location and resized local patches for videos on the Oulu-NPU dataset and the SiW dataset. The yellow boxes are the salient patches generated by the SAFPAD model. In every row, the first image indicates the input face image and the salient location. The following eight images are resized local patches. The top three rows are samples from the Oulu-NPU dataset. And the bottom three rows are from the SiW dataset. As for every three rows, they are respectively bona fide face, print attack, and replay attack.

From the visualization results, we can explain why these regions are salient. Fig. 13 illustrates that bona fide faces are located around the regions near the nose, eyes, or the boundary between face and background. The depth map of these regions changes more dramatically. The high-level feature encoding network will extract more discriminative information in these locations. Still, as for attack presentations, the SAFPAD model is more interested in the regions full of a variety of spoof clues, such as moire patterns, print artifacts, reflection artifacts, etc. As for the samples from the Oulu-NPU and SiW datasets, we observe that the salient region for attack presentations can be the edge of the print photos and the moire pattern from the electronic screen. The salient local feature encoding network will extract more discriminative feature of these clues. Hence, the regions around the cheek, the forehead of attack presentations and the edge of presentation attack instrument are usually located. This is similar to human perceptions, and humans also classify an image based on the spoofing cues hidden in the image. Further, these visualization results show the excellence of the deep reinforcement learning framework.

We can adopt the salient local information and global information of the samples to cluster them into different groups for classification. We utilized t-SNE [80] for dimension reduction to visualize the results. The t-SNE projects the fused feature to 2 dimensions and each group of 2 dimensions feature denotes one presentation attack instrument. Feature distribution of the testing videos on OULU-NPU Protocol 1 is shown in Fig. 14. The visual differences between bona fide faces and attack presentations tend to be very subtle, and it is difficult for humans to distinguish. We observe that generally the bona fide faces and attack presentations are not absolutely separated, despite there are few 2 dimensions feature clustered together with different labels. Actually, not only the bona fide faces and attack presentations, we can find that each presentation attack instrument is clustered. Consequently, it is possible to capture a presentation attack instruments cluster when training

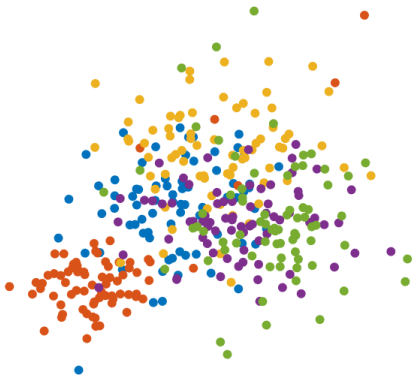

Fig. 14. Feature distribution visualization of the testing videos on OULU-NPU Protocol 1 using t-SNE. Color indicates: red=bona fide, green=printer1, blue=printer2, yellow=display1, purple=display2.

the SAFPAD model with presentation attack instrument labels. The result suggests that our proposed SAFPAD model has an excellent discrimination ability for distinguishing the bona fide faces and attack presentations.

V. Conclusions

In this paper, we have proposed a recurrent salience-aware reinforcement learning framework for face presentation attack detection, which aims to explore the salient local part information in the face image. In contrast to the existing approaches, we jointly train the SAFPAD model with the deep reinforcement learning framework. Specially, we adopt the multiscale feature fusion to guide the policy network to explore discriminative patches. Our methods achieve very competitive performance on five widely used face presentation attack detection datasets. Moreover, our approach improves the generalization capability for the complicated face presentation attack detection application conditions.

Acknowledgment

This research was partly supported by the National Key R&D Program of China (Grant No. 2020AAA0108302), Shenzhen Science and Technology Project (Grant No. JCYJ20200109143041798), and Beijing Academy of Artificial Intelligence (BAAI).

References

[1] Y. Atoum, Y. Liu, A. Jourabloo, and X. Liu, "Face anti-spoofing using patch and depth-based cnns," in *IJCB*, 2017, pp. 319–328.
[2] I. Chingovska and A. R. Dos Anjos, "On the use of client identity information for face antispoofing," *TIFS*, vol. 10, no. 4, pp. 787–796, 2015.
[3] A. Sepas-Moghaddam, F. Pereira, and P. L. Correia, "Light field-based face presentation attack detection: reviewing, benchmarking and one step further," *TIFS*, vol. 13, no. 7, pp. 1696–1709, 2018.
[4] J. Yang, Z. Lei, D. Yi, and S. Z. Li, "Person-specific face antispoofing with subject domain adaptation," *TIFS*, vol. 10, no. 4, pp. 797–809, 2015.
[5] A. Pinto, W. R. Schwartz, H. Pedrini, and A. de Rezende Rocha, "Using visual rhythms for detecting video-based facial spoof attacks," *TIFS*, vol. 10, no. 5, pp. 1025–1038, 2015.
[6] P. P. Chan, W. Liu, D. Chen, D. S. Yeung, F. Zhang, X. Wang, and C.-C. Hsu, "Face liveness detection using a flash against 2d spoofing attack," *TIFS*, vol. 13, no. 2, pp. 521–534, 2017.
[7] W. Kim, S. Suh, and J.-J. Han, "Face liveness detection from a single image via diffusion speed model," *TIP*, vol. 24, no. 8, pp. 2456–2465, 2015.
[8] S. R. Arashloo, J. Kittler, and W. Christmas, "Face spoofing detection based on multiple descriptor fusion using multiscale dynamic binarized statistical image features," *TIFS*, vol. 10, no. 11, pp. 2396–2407, 2015.
[9] 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," *TIFS*, vol. 10, no. 4, pp. 864–879, 2015.
[10] A. Agarwal, R. Singh, and M. Vatsa, "Face anti-spoofing using haralick features," in *BTAS*, 2016, pp. 1–6.
[11] Z. Boulkenafet, J. Komulainen, Z. Akhtar, A. Benlamoudi, D. Samai, S. E. Bekhouche, A. Ouafi, F. Dornaika, A. Taleb-Ahmed, L. Qin *et al.*, "A competition on generalized software-based face presentation attack detection in mobile scenarios," in *IJCB*, 2017, pp. 688–696.
[12] A. Jourabloo, Y. Liu, and X. Liu, "Face de-spoofing: Anti-spoofing via noise modeling," in *ECCV*, 2018, pp. 290–306.
[13] W. Bao, H. Li, N. Li, and W. Jiang, "A liveness detection method for face recognition based on optical flow field," in *ISPA*, 2009, pp. 233–236.
[14] Z. Boulkenafet, J. Komulainen, and A. Hadid, "Face spoofing detection using colour texture analysis," *TIFS*, vol. 11, no. 8, pp. 1818–1830, 2016.
[15] A. Pinto, S. Goldenstein, A. Ferreira, T. Carvalho, H. Pedrini, and A. Rocha, "Leveraging shape, reflectance and albedo from shading for face presentation attack detection," *TIFS*, vol. 15, pp. 3347–3358, 2020.
[16] T. de Freitas Pereira, A. Anjos, J. M. De Martino, and S. Marcel, "Lbp-top based countermeasure against face spoofing attacks," in *ACCV*, 2012, pp. 121–132.
[17] J. Komulainen, A. Hadid, and M. Pietikäinen, "Context based face anti-spoofing," in *BTAS*, 2013, pp. 1–8.
[18] J. Määttä, A. Hadid, and M. Pietikäinen, "Face spoofing detection from single images using micro-texture analysis," in *IJCB*, 2011, pp. 1–7.
[19] L. Li, X. Feng, Z. Boulkenafet, Z. Xia, M. Li, and A. Hadid, "An original face anti-spoofing approach using partial convolutional neural network," in *IPTA*, 2016, pp. 1–6.
[20] C. Nagpal and S. R. Dubey, "A performance evaluation of convolutional neural networks for face anti spoofing," in *IJCNN*, 2019, pp. 1–8.
[21] X. Qu, J. Dong, and S. Niu, "shallowcnn-le: A shallow cnn with laplacian embedding for face anti-spoofing," in *FG*, 2019, pp. 1–8.
[22] X. Yang, W. Luo, L. Bao, Y. Gao, D. Gong, S. Zheng, Z. Li, and W. Liu, "Face anti-spoofing: Model matters, so does data," in *CVPR*, 2019, pp. 3507–3516.
[23] T. de Freitas Pereira, A. Anjos, J. M. De Martino, and S. Marcel, "Can face anti-spoofing countermeasures work in a real world scenario?" in *ICB*, 2013, pp. 1–8.
[24] J. Yang, Z. Lei, S. Liao, and S. Z. Li, "Face liveness detection with component dependent descriptor," in *ICB*, 2013, pp. 1–6.
[25] K. Patel, H. Han, and A. K. Jain, "Secure face unlock: Spoof detection on smartphones," *TIFS*, vol. 11, no. 10, pp. 2268–2283, 2016.
[26] Z. Boulkenafet, J. Komulainen, and A. Hadid, "Face antispoofing using speeded-up robust features and fisher vector encoding," *SPL*, vol. 24, no. 2, pp. 141–145, 2016.
[27] D. C. Garcia and R. L. de Queiroz, "Face-spoofing 2d-detection based on moiré-pattern analysis," *TIFS*, vol. 10, no. 4, pp. 778–786, 2015.
[28] R. Shao, X. Lan, J. Li, and P. C. Yuen, "Multi-adversarial discriminative deep domain generalization for face presentation attack detection," in *CVPR*, 2019, pp. 10 023–10 031.
[29] X. Zhao, Y. Lin, and J. Heikkilä, "Dynamic texture recognition using volume local binary count patterns with an application to 2d face spoofing detection," *TMM*, vol. 20, no. 3, pp. 552–566, 2017.
[30] Z. Xu, S. Li, and W. Deng, "Learning temporal features using lstm-cnn architecture for face anti-spoofing," in *ACPR*, 2015, pp. 141–145.
[31] E. M. Nowara, A. Sabharwal, and A. Veeraraghavan, "Ppgsecure: Biometric presentation attack detection using photopletysmograms," in *FG*, 2017, pp. 56–62.
[32] S.-Q. Liu, X. Lan, and P. C. Yuen, "Remote photoplethysmography correspondence feature for 3d mask face presentation attack detection," in *ECCV*, 2018, pp. 558–573.
[33] X. Li, J. Komulainen, G. Zhao, P.-C. Yuen, and M. Pietikäinen, "Generalized face anti-spoofing by detecting pulse from face videos," in *ICPR*, 2016, pp. 4244–4249.
[34] S. Liu, P. C. Yuen, S. Zhang, and G. Zhao, "3d mask face anti-spoofing with remote photoplethysmography," in *ECCV*, 2016, pp. 85–100.
[35] X. Li, J. Chen, G. Zhao, and M. Pietikainen, "Remote heart rate measurement from face videos under realistic situations," in *CVPR*, 2014, pp. 4264–4271.
[36] Y. Liu, A. Jourabloo, and X. Liu, "Learning deep models for face anti-spoofing: Binary or auxiliary supervision," in *CVPR*, 2018, pp. 389–398.
[37] R. Raghavendra, K. B. Raja, and C. Busch, "Presentation attack detection for face recognition using light field camera," *TIP*, vol. 24, no. 3, pp. 1060–1075, 2015.
[38] A. George, Z. Mostaani, D. Geissenbuhler, O. Nikisins, A. Anjos, and S. Marcel, "Biometric face presentation attack detection with multi-channel convolutional neural network," *TIFS*, vol. 15, pp. 42–55, 2019.
[39] Z. Yu, Y. Qin, X. Li, Z. Wang, C. Zhao, Z. Lei, and G. Zhao, "Multi-modal face anti-spoofing based on central difference networks," in *CVPRW*, 2020, pp. 650–651.
[40] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski *et al.*, "Human-level control through deep reinforcement learning," *nature*, vol. 518, no. 7540, pp. 529–533, 2015.
[41] L. Xiao, Y. Li, G. Han, H. Dai, and H. V. Poor, "A secure mobile crowdsensing game with deep reinforcement learning," *TIFS*, vol. 13, no. 1, pp. 35–47, 2017.
[42] K. Arulkumaran, M. P. Deisenroth, M. Brundage, and A. A. Bharath, "Deep reinforcement learning: A brief survey," *SPM*, vol. 34, no. 6, pp. 26–38, 2017.
[43] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[44] D. Silver, G. Lever, N. Heess, T. Degris, D. Wierstra, and M. Riedmiller, "Deterministic policy gradient algorithms," in *ICML*, 2014.
[45] D. Silver, A. Huang, C. J. Maddison, A. Guez, L. Sifre, G. Van Den Driessche, J. Schrittwieser, I. Antonoglou, V. Panneershelvam, M. Lanctot *et al.*, "Mastering the game of go with deep neural networks and tree search," *nature*, vol. 529, no. 7587, pp. 484–489, 2016.
[46] Y.-H. Ho, C.-Y. Cho, W.-H. Peng, and G.-L. Jin, "Sme-net: Sparse motion estimation for parametric video prediction through reinforcement learning," in *ICCV*, 2019, pp. 10 462–10 470.
[47] W. Wu, D. He, X. Tan, S. Chen, and S. Wen, "Multi-agent reinforcement learning based frame sampling for effective untrimmed video recognition," in *ICCV*, 2019, pp. 6222–6231.
[48] R. Cai, H. Li, S. Wang, C. Chen, and A. C. Kot, "Drl-fas: a novel framework based on deep reinforcement learning for face anti-spoofing," *TIFS*, vol. 16, pp. 937–951, 2020.
[49] S. Mathe, A. Pirinen, and C. Sminchisescu, "Reinforcement learning for visual object detection," in *CVPR*, 2016, pp. 2894–2902.
[50] Y. Wang, M. Dong, J. Shen, Y. Wu, S. Cheng, and M. Pantic, "Dynamic face video segmentation via reinforcement learning," in *CVPR*, 2020, pp. 6959–6969.
[51] S. Yun, J. Choi, Y. Yoo, K. Yun, and J. Young Choi, "Action-decision networks for visual tracking with deep reinforcement learning," in *CVPR*, 2017, pp. 2711–2720.
[52] Z. Yang, L. Huang, Y. Chen, Z. Wei, S. Ahn, G. Zelinsky, D. Samaras, and M. Hoai, "Predicting goal-directed human attention using inverse reinforcement learning," in *CVPR*, 2020, pp. 193–202.
[53] J. Shao, Y. Jiang, G. Wang, Z. Li, and X. Ji, "Pfrl: Pose-free reinforcement learning for 6d pose estimation," in *CVPR*, 2020, pp. 11 454–11 463.
[54] Q. Cao, L. Lin, Y. Shi, X. Liang, and G. Li, "Attention-aware face hallucination via deep reinforcement learning," in *CVPR*, 2017, pp. 690–698.
[55] C. N. Duong, K. Luu, K. G. Quach, N. Nguyen, E. Patterson, T. D. Bui, and N. Le, "Automatic face aging in videos via deep reinforcement learning," in *CVPR*, 2019, pp. 10 013–10 022.
[56] M. Wang and W. Deng, "Mitigating bias in face recognition using skewness-aware reinforcement learning," in *CVPR*, 2020, pp. 9322–9331.
[57] K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using rnn encoder-decoder for statistical machine translation," *arXiv preprint arXiv:1406.1078*, 2014.
[58] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *CVPR*, 2016, pp. 770–778.
[59] Y. Feng, F. Wu, X. Shao, Y. Wang, and X. Zhou, "Joint 3d face reconstruction and dense alignment with position map regression network," in *ECCV*, 2018, pp. 534–551.
[60] R. J. Williams, "Simple statistical gradient-following algorithms for connectionist reinforcement learning," *Machine Learning*, vol. 8, no. 3-4, pp. 229–256, 1992.
[61] I. Chingovska, A. Anjos, and S. Marcel, "On the effectiveness of local binary patterns in face anti-spoofing," in *BIOSIG*, 2012, pp. 1–7.
[62] Z. Zhang, J. Yan, S. Liu, Z. Lei, D. Yi, and S. Z. Li, "A face antispoofing database with diverse attacks," in *ICB*, 2012, pp. 26–31.
[63] Z. Boulkenafet, J. Komulainen, L. Li, X. Feng, and A. Hadid, "Oulu-npu: A mobile face presentation attack database with real-world variations," in *FG*, 2017, pp. 612–618.
[64] Z. Mostaani, A. George, G. Heusch, D. Geissbuhler, and S. Marcel, "The high-quality wide multi-channel attack (hq-wmca) database," *arXiv preprint arXiv:2009.09703*, 2020.
[65] I. Standard, "Information technology–biometric presentation attack detection–part 3: testing and reporting," *ISO: Geneva, Switzerland*, 2017.
[66] "Livdet competition," https://livdet.org/competitions.php, accessed on 2021-10-01.
[67] V. Nair and G. E. Hinton, "Rectified linear units improve restricted boltzmann machines," in *ICML*, 2010, pp. 807–814.
[68] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," *arXiv preprint arXiv:1502.03167*, 2015.
[69] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.
[70] C. Lin, Z. Liao, P. Zhou, J. Hu, and B. Ni, "Live face verification with multiple instantialized local homographic parameterization." in *IJCAI*, 2018, pp. 814–820.
[71] L. Li, Z. Xia, A. Hadid, X. Jiang, H. Zhang, and X. Feng, "Replayed video attack detection based on motion blur analysis," *TIFS*, vol. 14, no. 9, pp. 2246–2261, 2019.
[72] H. Chen, G. Hu, Z. Lei, Y. Chen, N. M. Robertson, and S. Z. Li, "Attention-based two-stream convolutional networks for face spoofing detection," *TIFS*, vol. 15, pp. 578–593, 2019.
[73] "Detware," https://www.nist.gov/itl/iad/mig/tools, accessed on 2021-10-01.
[74] A. George and S. Marcel, "Deep pixel-wise binary supervision for face presentation attack detection," in *ICB*. IEEE, 2019, pp. 1–8.
[75] George, Anjith and Marcel, Sébastien, "Learning one class representations for face presentation attack detection using multi-channel convolutional neural networks," *TIFS*, vol. 16, pp. 361–375, 2020.
[76] A. Parkin and O. Grinchuk, "Recognizing multi-modal face spoofing with face recognition networks," in *CVPRW*, 2019, pp. 0–0.
[77] S. Bharadwaj, T. I. Dhamecha, M. Vatsa, and R. Singh, "Computationally efficient face spoofing detection with motion magnification," in *CVPRW*, 2013, pp. 105–110.
[78] A. Pinto, H. Pedrini, W. R. Schwartz, and A. Rocha, "Face spoofing detection through visual codebooks of spectral temporal cubes," *TIP*, vol. 24, no. 12, pp. 4726–4740, 2015.
[79] J. Yang, Z. Lei, and S. Z. Li, "Learn convolutional neural network for face anti-spoofing," *arXiv preprint arXiv:1408.5601*, 2014.
[80] L. v. d. Maaten and G. Hinton, "Visualizing data using t-sne," *JMLR*, vol. 9, no. Nov, pp. 2579–2605, 2008.

Bingyao Yu received the B.S. degree in the Department of Automation, Tsinghua University, China, in 2018. He is currently a Ph.D. Candidate with the Department of Automation, Tsinghua University, China. His current research interests include computer vision, deep learning, and face anti-spoofing. He serves as a regular reviewer member for a number of journals and conferences, e.g. TIP, TBIOM, TCSVT, PR, ICME, and ICIP.

Jiwen Lu (M'11-SM'15) received the B.Eng. degree in mechanical engineering and the M.Eng. degree in electrical engineering from the Xi'an University of Technology, Xi'an, China, in 2003 and 2006, respectively, and the Ph.D. degree in electrical engineering from Nanyang Technological University, Singapore, in 2012. He is currently an Associate Professor with the Department of Automation, Tsinghua University, Beijing, China. His current research interests include computer vision, pattern recognition, and machine learning. He was/is a member of the Image, Video and Multidimensional Signal Processing Technical Committee, Multimedia Signal Processing Technical Committee and the Information Forensics and Security Technical Committee of the IEEE Signal Processing Society, a member of the Multimedia Systems and Applications Technical Committee and the Visual Signal Processing and Communications Technical Committee of the IEEE Circuits and Systems Society, respectively. He serves as the General Co-Chair for the International Conference on Multimedia and Expo (ICME) 2022, the Program Co-Chair for the International Conference on Multimedia and Expo 2020, the International Conference on Automatic Face and Gesture Recognition (FG) 2023, and the International Conference on Visual Communication and Image Processing (VCIP) 2022. He serves as the Co-Editor-of-Chief for Pattern Recognition Letters, an Associate Editor for the IEEE Transactions on Image Processing, the IEEE Transactions on Circuits and Systems for Video Technology, and the IEEE Transactions on Biometrics, Behavior, and Identity Sciences. He was a recipient of the National Natural Science Funds for Distinguished Young Scholar. He is a Fellow of IAPR.

Xiu Li received the Ph.D. degree in computer integrated manufacturing from the Nanjing University of Aeronautics and Astronautics in 2000. Since then, she has been with Tsinghua University, Beijing, China. Her research interests include intelligent system, pattern recognition, and data mining.

Jie Zhou (M'01-SM'04) received the BS and MS degrees both from the Department of Mathematics, Nankai University, Tianjin, China, in 1990 and 1992, respectively, and the PhD degree from the Institute of Pattern Recognition and Artificial Intelligence, Huazhong University of Science and Technology (HUST), Wuhan, China, in 1995. From then to 1997, he served as a postdoctoral fellow in the Department of Automation, Tsinghua University, Beijing, China. Since 2003, he has been a full professor in the Department of Automation, Tsinghua University. His research interests include computer vision, pattern recognition, and image processing. In recent years, he has authored more than 100 papers in peer-reviewed journals and conferences. Among them, more than 30 papers have been published in top journals and conferences such as the IEEE Transactions on Pattern Analysis and Machine Intelligence, IEEE Transactions on Image Processing, and CVPR. He is an associate editor for the IEEE Transactions on Pattern Analysis and Machine Intelligence and two other journals. He received the National Outstanding Youth Foundation of China Award. He is a senior member of the IEEE.