Multi-Domain Incremental Learning for Face Presentation Attack Detection

Keyao Wang*[1], Guosheng Zhang*[1], Haixiao Yue*[1], Ajian Liu†[2], Gang Zhang[1], Haocheng Feng[1], Junyu Han[1], Errui Ding[1], Jingdong Wang[1]

[1]Department of Computer Vision Technology (VIS), Baidu Inc
[2]CBSR&MAIS, Institute of Automation, Chinese Academy of Sciences (CASIA)
{wangkeyao, zhangguosheng, yuehaixiao, zhanggang03, fenghaocheng, hanjunyu, dingerrui}@baidu.com,
ajianliu92@gmail.com, wangjingdong@outlook.com

Abstract

Previous face Presentation Attack Detection (PAD) methods aim to improve the effectiveness of cross-domain tasks. However, in real-world scenarios, the original training data of the pre-trained model is not available due to data privacy or other reasons. Under these constraints, general methods for fine-tuning single-target domain data may lose previously learned knowledge, leading to the issue of catastrophic forgetting. To address these issues, we propose a Multi-Domain Incremental Learning (MDIL) method for PAD, which not only learns knowledge well from the new domain but also maintains the performance of previous domains stably. To this end, we propose an Adaptive Domain-specific Experts (ADE) framework based on the vision transformer to preserve the discriminability of previous domains. Moreover, we present an asymmetric classifier to keep the output distribution of different classifiers consistent, thereby improving the generalization ability. Extensive experiments show that our proposed method achieves state-of-the-art performance compared to prior methods of incremental learning. Excitingly, under more stringent setting conditions, our method approximates or even outperforms DA/DG-based methods.

Introduction

In recent years, face recognition (FR) techniques have been widely exploited in different application scenarios, such as smartphone login, financial payment, access control, etc. While FR systems may suffer from various presentation attacks (PAs), e.g., printed photos, video replay, and 3D masks, which seriously threaten the credibility of facial information and bring great challenges to public security management. To address these issues, various PAD methods have been proposed, including the hand-crafted methods (Freitas Pereira et al. 2012; Patel, Han, and Jain 2016) and the deep learning methods based on auxiliary supervision (Atoum et al. 2017; Liu et al. 2019).

Although these methods have achieved promising results under intra-dataset scenarios, they neglect domain discrepancy across different domains and may encounter performance degradation when adapting to new domains. To mitigate this problem of domain shift, recent studies introduce domain adaptation (DA) (Jia et al. 2021; Wang et al. 2019) and domain generalization (DG) (Zhou et al. 2021; Chen et al. 2021) into the field of PAD. As shown in Figure 1, DA-based methods focus on transferring performance on the unlabeled target domain, but they may affect the performance of the source domain. In contrast, DG-based methods aim to achieve out-of-distribution generalization by using multi-source domain data for model learning simultaneously.

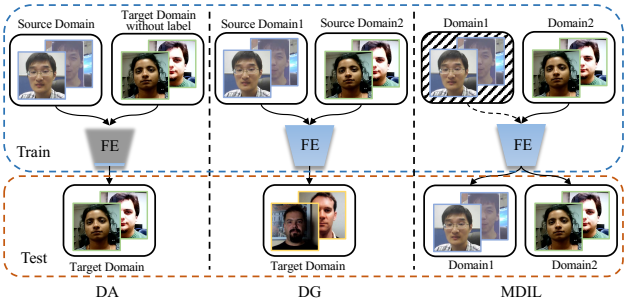

Figure 1: The comparison of different methods. DA-based methods align the unlabeled target domain with source domains. DG-based methods eliminate dependency on the target domain and test directly on the unseen target domain. MDIL-based methods aim to obtain satisfactory results from previous domain data without revisiting old domain data.

In general, DA/DG-based methods address the domain transfer problem. But in most practical scenarios, part or all of the data in the source domain cannot be accessed by data privacy. MDIL-based methods solve the problem of mitigating the catastrophic forgetting of the original domain data information when only using the target domain data for training. As shown in the third column in Figure 1, MDIL aims to train a single model on sequential non-stationary domains without revisiting the previous domain data. In practical PAD application, since an initially trained model fails to identify novel attack types from unknown domains, a straightforward way to solve this problem is to retrain the model from scratch on both old and new domain data. However, model retraining is computationally costly because it requires storing large amounts of previous domain

*These authors contributed equally.
†Corresponding author

data. Therefore, it is likely that MDIL may be more suitable for PAD scenarios with a potential domain transfer, as it can reduce the data storage requirements of model retraining.

However, there are two main challenges with MDIL-PAD. (1) **Domain gap.** Due to the diversity of data collection environments and the ongoing emergence of novel attack types, domain incremental learning task for PAD simultaneously couples the challenges of domain incremental learning (DIL) and class incremental learning (CIL). Larger domain gaps lead to more severe forgetting. (2) **Domain agnostic.** A common and practical constraint in DIL is that domain indexes are not provided for inference. So we can not tackle this problem by training a separate, independent model for each domain. L2P(Wang et al. 2022b) and S-Prompts(Wang, Huang, and Hong 2022) discarded the need for task indexes by designing a query-key mechanism or clustering strategy to automatically select relevant prompts for each instance. However, these strategies involve additional computational overhead. They will be discussed in appendix B.

To this end, we propose a novel multi-domain incremental framework for PAD that leverages the multi-experts to instruct the model adaptively. In our method, the Adaptive Domain-specific Experts (ADE) blocks maintain the isolation of domain-specific parameters and the sharing of domain-invariant parameters, which is beneficial for mitigating catastrophic forgetting. Moreover, we first design a flexible Instance-wise Router (IwR) module for selecting the relevant expert. During inference, the domain-agnostic instances can obtain the associated domain index based on the similarity with the domain centers. Then, the appropriate index guides the instance into the ADE blocks with the corresponding expert branch by gating mechanism. Considering the feature separation of spoof samples among different domains, we consolidate the multi-classifiers into a unified asymmetric classifier. This design helps to keep the consistency of the predicted probability distribution (PPD) of live samples from different classifiers. Furthermore, the proposed method can adapt to the unknown domain and operate for the dynamically added domains in end-to-end training.

- We propose an adaptive domain-specific experts framework for PAD in an incremental update pattern, which can maintain satisfactory results in both previous and new domains. Besides, an innovative IwR is designed to deal with domain-agnostic instances.
- Considering the sparsity and discreteness of spoof samples, an asymmetric classifier is designed to solve the problem of PPD of live samples inconsistent in open appending domains.
- Extensive experiments are conducted on widely used benchmark datasets, which demonstrate the effectiveness of the proposed method and also illustrate that our method achieves state-of-the-art performance.

Related Work

Face Presentation Attack Detection

Face PAD aims to detect spoof attacks of various types and improve the security of face recognition systems. With the development of deep learning, many researchers (Wang et al. 2020b; Zhang et al. 2021) utilize the convolutional neural network to enhance the extraction ability of face representation. Considering the lack of sufficient supervision, some approaches (Liu, Jourabloo, and Liu 2018; Liu et al. 2019) design different auxiliary supervisions to improve the performance of classification, such as depth map (Yu et al. 2021), reflection map (Kim et al. 2019), and rPPG signals (Hu et al. 2021). These works achieve promising results with intra-data but neglect the domain gap across different domains.

To achieve better generalized performance in the target data, DA (Jia et al. 2021; Li et al. 2018; Wang et al. 2019, 2020a) and DG (Chen et al. 2021; Jia et al. 2020; Liu et al. 2021a,b; Shao et al. 2019) are introduced into the PAD area. SDA (Wang et al. 2021a) designs a domain adaptor to utilize the unlabeled test domain data at inference. GDA (Zhou et al. 2022b) stylizes the unlabeled target data to the source-domain style via image translation for feature alignment. In contrast, DG aims to learn a generalized representation from multi-source domains, independent of the target domain. SSDG (Jia et al. 2020) leverages asymmetric triplet loss and adversarial learning to regulate live samples and distinguish spoof samples from source domains. SSAN (Wang et al. 2022a) designs style assembly layers to combine indistinguishable content features and domain-specific style features.

Nevertheless, the above methods can effectively narrow the gap between the source domain and the target domain, they only focus on the transfer performance of the target domain and neglect the source domain.

Incremental Learning

In the application of multiple domains, the training data of the original domain are generally inaccessible due to data privacy, and when learning a new domain, the model may catastrophically forget what it learned previously(Kirkpatrick et al. 2017). Incremental Learning (IL) is introduced to alleviate the problem of the long-studied pattern. There are mainly three categories: regularization-based, replay-based, and parameter isolation-based methods. Regularization-based methods (Zenke, Poole, and Ganguli 2017; Aljundi et al. 2018) can consolidate these weights of previous tasks according to their importance. Replay-based techniques store previous experience by implicitly generating replays (Chenshen et al. 2018; Ostapenko et al. 2019) or explicitly displaying original samples (Hou et al. 2019; Wu et al. 2019) to preserve the representation ability of previous domains. As for parameter isolation methods (Mallya and Lazebnik 2018; Rebuffi, Bilen, and Vedaldi 2018), allocating specific model parameters to each task maintains maximal stability by fixing the parameter subsets of previous tasks. LwF (Li and Hoiem 2017) leverages knowledge distillation to maintain the performance of older tasks after adding new tasks. Inspired by the VPT (Jia et al. 2022), some prompting methods (Douillard et al. 2022; Wang, Huang, and Hong 2022) are proposed to alleviate the performance degradation of previous domains by introducing a small number of parameters. L2P (Wang et al. 2022b) utilizes the prompt pool to store encoded knowledge and

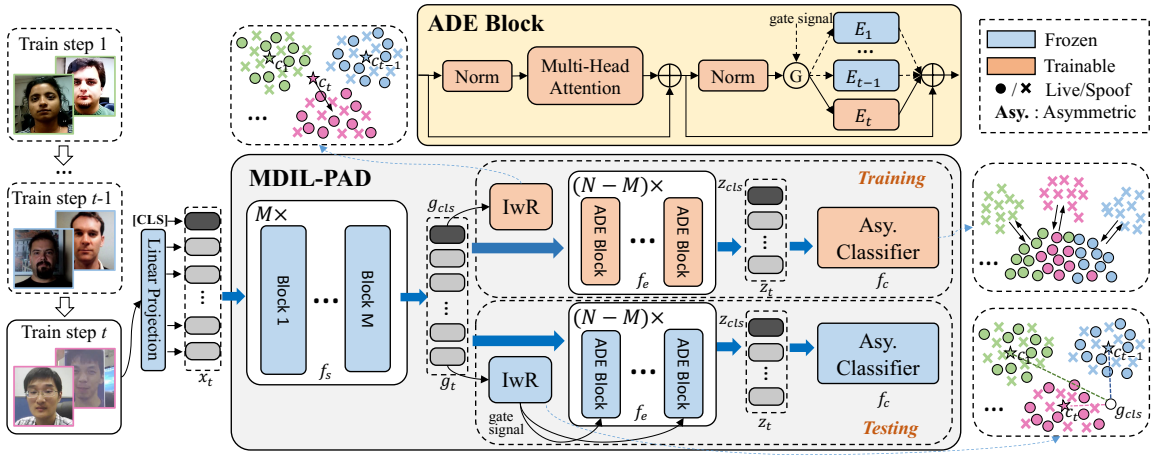

Figure 2: The overall framework of our proposed method. Our network is divided into the first M standard vision transformer blocks as shared encoder f_s and the last $(N-M)$ ADE blocks as expert decoder f_e. The processing steps are as follows: 1) Images from t-th domain are first embedded to tokens and then processed by the f_s for generating $\boldsymbol{g}_t$. 2) The first embedding of $\boldsymbol{g}_{cls}$ is fed to IwR for learning the domain center $\boldsymbol{c}_t$ in the training phase and predicting the gate signal to choose the appropriate expert branch in ADE blocks at test time. 3) The result is predicted by the proposed asymmetric classifier network f_c.

presumes the prompt tokens into the input tokens. In the terms of PAD, the article (Pérez-Cabo et al. 2020) introduces an IL framework into PAD for the first time, which follows the few-shot learning paradigm. For novel face spoof attack types, the method (Rostami et al. 2021) is proposed to treat them as anomalies and correctly classify them via experience replay. (Guo et al. 2022) propose the FAS-wrapper, which employs a regularization-based approach to facilitate knowledge transfer from pre-trained models for MDL. (Cai et al. 2023) propose the rehearsal-free method for Domain Continual Learning of FAS, which deals with catastrophic forgetting and unseen DG problems simultaneously.

Unlike these works, we design a buffer-free dynamic IL framework containing the learnable domain-specific information to mitigate the catastrophic forgetting of previous domains. Furthermore, we propose a multi-expert framework for MDIL, which preserves the parameter independence to learn the corresponding domain knowledge adaptively.

Methodology

In multi-domain incremental learning, $\mathcal{T}$ domains are presented sequentially, defined as $\mathcal{D} = \{(\mathcal{D}_1, \cdots, \mathcal{D}_T)\}$. Learning follows incremental steps, where each step involves learning an existing model for the current domain. The t-th input domain is defined as $\mathcal{D}_t = \{(\boldsymbol{x}_t^i, y_t^i)\}_{i=1}^{n_t}$, where $\boldsymbol{x}_t^i \in \mathcal{X}$ represents the input sample, $y_t^i \in \mathcal{Y}$ is the corresponding label, and n_t is the number of samples in $\mathcal{D}_t$. We aim to train a single PAD model $\mathcal{M}(\mathcal{X}) = \mathcal{Y}$, that predicts $y = \mathcal{M}(x)$ for any test sample x. At any learning step t, data from previous domains $\{(\mathcal{D}_1, \cdots, \mathcal{D}_{t-1})\}$ are not available for training.

Overview

We propose the Adaptive Domain-specific Experts (ADE) framework for coping with the performance degradation in previous domains, As shown in Figure 2, it consists of three key components: the ADE block with multiple experts, the Instance-wise Router (IwR), and an asymmetric classifier.

Specifically, we define an image sample from datasets as $\boldsymbol{x} \in \mathbb{R}^{H \times W \times C}$, where H, W represent the length and width of the image, respectively, and C is the number of channels. The image is first reshaped and split into a sequence of 2D patches $\boldsymbol{x}_p \in \mathbb{R}^{L \times S^2 \times C}$, where S is the side length of each image patch and $L = HW/S^2$ represents the number of patches. At the start of training, the sequence of embeddings $\boldsymbol{x}_p$ are first fed into the linear projection to generate the image tokens $\boldsymbol{x}_p \in \mathbb{R}^{L \times S^2 \times C} \rightarrow \boldsymbol{x}_t \in \mathbb{R}^{(L+1) \times D}$, where D is the embedding dimension, and the extra dimension on L is the corresponding class token.

Based on ViT (Dosovitskiy et al. 2020), our network consists of total N transformer blocks. We divide it into the first M blocks as shared encoder f_s and the last $(N-M)$ blocks as expert decoder f_e, while the whole f_e are designed by ADE blocks. Next, the image tokens $\boldsymbol{x}_t$ from t-th domain are sent to the shared encoder f_s with parameters W_{s_1}:

$$\boldsymbol{g}_t = f_s(\boldsymbol{x}_t; W_{s_1}), \tag{1}$$

the first embedding in $\boldsymbol{g}_t \in \mathbb{R}^{(L+1) \times D}$ is defined as $\boldsymbol{g}_{cls} \in \mathbb{R}^D$. Then, we propose the IwR module to predict the *gate signal* to choose the most appropriate domain-specific expert branch for ADE blocks when testing the domain-agnostic instance. And we update ADE blocks f_e with parameters W_{s_2}, W_{e_t} of the corresponding expert branch, which keeps the different domains of knowledge independent.

$$\boldsymbol{z}_t = f_e(\boldsymbol{g}_t, t; W_{s_2}, W_{e_t}), \tag{2}$$

where $\boldsymbol{z}_t \in \mathbb{R}^{(L+1) \times D}$ is the output feature of the t-th expert branch. After getting the feature $\boldsymbol{z}_t$, we split the first embedding of $\boldsymbol{z}_t$ as $\boldsymbol{z}_{cls} \in \mathbb{R}^D$. The final result $\boldsymbol{p}_t$ is generated by the proposed asymmetric classifier network f_c with

Algorithm 1: The Procedure of MDIL-PAD.

Require: Sequential domain dataset $\mathcal{D}_t = \{(\boldsymbol{x}_i^t, y_i^t)\}_{i=1}^{n_t}$.
\\Training:
for $t = 1, \cdots, T$ **do**
  Reload W_{s_1}, W_{s_2}, θ_l from $t-1$ step. Random initialize W_{e_t}, c_t, θ_{s_t}. Freeze W_{s_1}.
  for $i = 1, \cdots, MaxEpoches$ **do**
    Forward pass $\boldsymbol{x}_t$ via W_{s_1}, Eq. 1
    Compute domain-specific loss L_{gate} by Eq. 5
    Forward pass $\boldsymbol{g}_t$ via W_{s_2}, W_{e_t}, Eq. 2
    Forward pass $\boldsymbol{z}_{cls}$ by θ_l, θ_{s_t}, Eq. 3
    Compute classifier loss L_{cls} by Eq. 6
    Compute total loss via L_{all} Eq. 7
  end for
  Discard training data $\mathcal{D}_t$.
end for
\\Inference:
Forward pass $\boldsymbol{x}_t$ via W_{s_1}, Eq. 1
Forward pass $\boldsymbol{g}_{cls}$ via trained Router f_r, Eq. 8
Forward pass $\boldsymbol{g}_t$ via W_{s_2}, W_{e_t}, Eq. 2
Forward pass $\boldsymbol{z}_{cls}$ via θ_l, Eq. 3

parameters θ_l, θ_{s_t}:

$$\boldsymbol{p}_t = f_c(\boldsymbol{z}_{cls}, t; \theta_l, \theta_{s_t}). \tag{3}$$

Note that when training on different domains, we keep the shared encoder f_s frozen, only fine-tuning the expert decoder f_e and the asymmetric classifier network f_c.

Adaptive Domain-Specific Experts

To mitigate the performance degradation of previous domains, we design the domain-specific experts framework. This framework can keep the parameter independent among different domains while sharing the knowledge between similar domains. Besides, the proposed framework can adaptively deal with the domain-agnostic sample in the inference stage. In this section, we describe the designed ADE blocks and the instance-wise router in detail. The whole process of the proposed framework is described in Algorithm 1.

ADE Block. Compared to the original ViT blocks, we dynamically extend the original MLPs to multi domain-wised experts, which have the same architecture $E_t(x) = MLP(x)$. Specifically, we define the sharing of parameters and the t-th expert parameters in the ADE block as W_{s_2}, W_{e_t}, respectively. This design decouples the network parameters into a set of domain-invariant and domain-specific parameters. Thus, they can learn domain-specific knowledge separately without interference while leveraging domain-invariant knowledge to consolidate the generalization capability and alleviate catastrophic forgetting. Equipped with the learnable gating mechanism, the proposed ADE block can adaptively choose the domain-related expert in inference. The output $\boldsymbol{z}_t$ is obtained by Eq. 2.

When learning a new domain $\mathcal{D}_{t+1}$ in step $t+1$, we keep the previously trained parameters $W_{e_j} (1 \le j \le t)$ frozen and update the sharing of parameters W_{s_2} and the newly added domain parameters $W_{e_{t+1}}$ according to the process:

$$\boldsymbol{z}_{t+1} = f_e(\boldsymbol{g}_{t+1}, t+1; W_{s_2}, W_{e_{t+1}}). \tag{4}$$

Note that the domain index(gate signal) t is known during the training phase, while it is unknown at test time.

Instance-Wise Router. As mentioned above, we employ isolated experts to maintain knowledge independence across different domains, which brings up a critical question: how to automatically choose domain-related experts during the inference phase? To deal with this problem, we design a learnable gating mechanism, Instance-wise Router (IwR), to assign each domain-agnostic instance to a domain-related expert branch.

During the training phase on the current incremental domain $\mathcal{D}_t$, we aim to find a domain center that aligns with the latent feature $\boldsymbol{g}_{cls} \in \mathbb{R}^D$. We assume that images from the same domain have a similar distribution on the high-level feature space projected by a well-pretrained network. Specifically, we denote the domain center as $\boldsymbol{c}_t \in \mathbb{R}^D$ and the instance-wise router network as f_r. As shown in Figure 2, the corresponding domain center $\boldsymbol{c}_t$ is expected to be close to the feature center of the current domain. Thus, we achieve this goal by minimizing a cross-entropy loss:

$$\mathcal{L}_{\text{gate}} = -\frac{1}{n_t} \sum_{i=1}^{n_t} log \frac{e^{sim(\boldsymbol{g}_{cls}^i, \boldsymbol{c}_t)/\tau}}{1 - e^{sim(\boldsymbol{g}_{cls}^i, \boldsymbol{c}_t)/\tau}}, \tag{5}$$

where n_t is the number of t-th domain images, and τ is a temperature coefficient. Besides, $sim(\boldsymbol{a}, \boldsymbol{b}) = \frac{\boldsymbol{a} \cdot \boldsymbol{b}}{\|\boldsymbol{a}\| \|\boldsymbol{b}\|}$ is the cosine similarity calculation.

Asymmetric Classifier

Considering the design of the classifier for multi-branch network architecture, one way to avoid catastrophic forgetting is to construct multiple separate, independent classifiers $\{[\theta_{l_1}, \theta_{s_1}], \cdots, [\theta_{l_T}, \theta_{s_T}]\}$, and each classifier corresponds to a domain-related expert branch. However, outputs of the PAD model are commonly expected to be predicted probability of live $\boldsymbol{p}_l = f_c(\boldsymbol{z}_{cls}, t; \theta_{l_t})$, differentiated by a threshold. However, this approach makes it hard to ensure consistency of the predicted probability distribution (PPD) across different classifiers. Thus, the setting of using the same threshold for multiple classifiers easily causes poor performance. Another alternative approach is to design a shared single classifier. Although it gets rid of inconsistencies of PPD, it still suffers from catastrophic forgetting.

Inspired by (Jia et al. 2020), the feature distribution of the live samples is compact while that of the spoof samples present is dispersed across domains. It suggests that the domain gap is mainly reflected in the spoof samples. Therefore, we consolidate the multi-classifiers into a unified asymmetric classifier. Specifically, we use a shared class center θ_l for live samples across different domains, while the spoof samples from different domains are regarded as an individual category. As shown in Figure 3, the asymmetric classifier $\{\theta_l, \theta_{s_1}, \cdots, \theta_{s_T}\}$ contains $T+1$ categories. The predicted probability of live $\boldsymbol{p}_l = f_c(\boldsymbol{z}_{cls}; \theta_l)$ is domain-independent, and the PPD significantly consistent due to the unified decision boundaries. For training data with domain index t, the output can be formulated by Eq. 3.

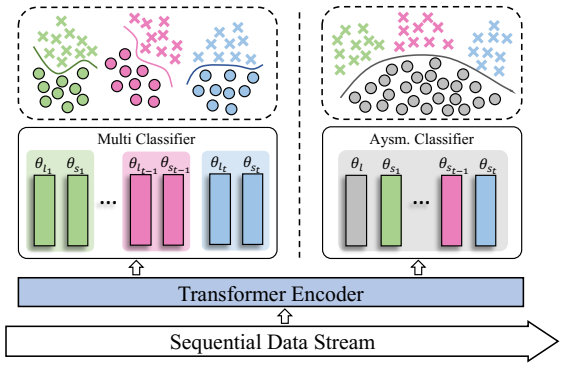

Figure 3: The architecture of the asymmetric classifier.

The first category refers to the live face, and the other categories are face spoof samples of various domains. The unified classifier f_c is optimized by the cross-entropy loss function:

$$\mathcal{L}_{\text{cls}} = -\frac{1}{n_t}\sum_{i=1}^{n_t} y_t^i log(\boldsymbol{p}_t^i) + (1-y_t^i)log(1-\boldsymbol{p}_t^i). \quad (6)$$

The total training loss function contains two parts:

$$\mathcal{L}_{\text{all}} = \mathcal{L}_{\text{cls}} + \alpha\mathcal{L}_{\text{gate}}, \quad (7)$$

where α is a scaling factor and $\mathcal{L}_{\text{gate}}$ set to 1.0 for all experiments. The proposed algorithm is optimized by continuously fine-tuning new domain data.

Inference Phase

The proposed method can adaptively solve the domain-agnostic incremental problem. Furthermore, it can be performed for the open appending domains in the end-to-end network. For a query domain-agnostic instance $\boldsymbol{x}$, we obtain the feature $\boldsymbol{g}_{cls} \in \mathbb{R}^D$ from pretrained shared encoder f_s. Then we calculate the feature similarity between $\boldsymbol{g}_{cls}$ and well-trained domain centers $\boldsymbol{c}_i$ in IwR. The domain index with the highest similarity is selected as the gate signal:

$$t = \underset{i\subseteq[1,T]}{argmax} f_r(\boldsymbol{g}_{cls}; \boldsymbol{c}_i). \quad (8)$$

According to the predicted domain index t, we perform the corresponding expert branch on embedding to generate the final output of face prediction.

Experiment

Datasets. We evaluate the effectiveness of our method on five PAD datasets: OULU-NPU (Boulkenafet et al. 2017) (O for short), CASIA-MFSD (Zhang et al. 2012) (C for short), Idiap ReplayAttack (Chingovska, Anjos, and Marcel 2012) (I for short), MSU-MFSD (Wen, Han, and Jain 2015) (M for short), and SiW (Liu, Jourabloo, and Liu 2018) (S for short). These datasets contain face samples from different capture devices, illumination, background, and spoof attack types, which results in great distribution discrepancies.

Implementation Details. We train our method using the Stochastic Gradient Descent (SGD) optimizer with a momentum of 0.9, an initial learning rate of 0.01, and a batch size of 48. Input images are resized to 224×224. To compare fairly, we adopt the same network ViT-B/16 (Dosovitskiy et al. 2020) across all methods in Table 1. Specifically, we reimplement LwF (Li and Hoiem 2017) by utilizing the ViT. For DyTox (Douillard et al. 2022) and S-iPr(Wang, Huang, and Hong 2022), we use their official implementations by tuning their block number to 12 since the original paper set it to 6, and we maintain the same settings for all experiments.

Evaluation Metrics. Following the work of (Jia et al. 2020), we utilize the Half Total Error Rate (HTER) and Area Under Curve (AUC) to evaluate the performance. To quantify the overall performance, similar to (Kanakis et al. 2020), we define the average decrease in HTER of each task t with respect to the multi-domain performance b as $\Delta m\%$. For the model q, the $\Delta m\%$ is calculated by:

$$\Delta m\% = \frac{1}{T}\sum_{t=1}^{T}\frac{HTER_{b,t} - HTER_{q,t}}{1 - HTER_{b,t}}. \quad (9)$$

Lower $\Delta m\%$ indicates less performance drop in previous domain.

Comparison to the State-of-the-Art Methods

To validate the effectiveness of our proposed approach, we compare it with other IL baselines. Joint training (JT) is trained with all domain datasets and considered an upper-bound performance. Fine-tuning (FT) is optimized on the new domain without any explicit effort to mitigate forgetting, which is considered a lower-bound performance. Feature extraction (FE) freezes all backbone parameters and just trains the fully connected (FC) layer of a new domain.

Results of 4-Domain Incremental Settings. In this section, we first construct a model on dataset M in step 1. Then the same model is optimized by dataset C in step 2. Subsequently, we fine-tune the model on new datasets with the domain shift I and O in steps 3 and 4 and evaluate the results from previous domains. As shown in Table 1, we make the following observations. (1) Compared with other general methods, the proposed approach effectively mitigates the performance degradation of previous domains. Based on FT, our method obtains significant improvements of 15.40% and 18.83% on dataset M in step 2 and step 3, respectively. In addition, the newly supplemented domain in different steps also maintains optimal performance. (2) Compared with other IL-based methods, our method achieves the minimum average drop $\Delta m\%$ on previous domains under various steps. And in longer steps like steps 3 and 4, our method outperforms other methods, showing that our proposed method can retain different domains of knowledge more persistently by constructing domain-specific experts.

Results of Cross-Attack-Type Incremental Settings. In this section, we focus on the larger domain gap caused by the absence of overlapping attack types in sequence domains. Specifically, we exclude Print/Replay type attacks from SiW

Methods	Step 2 (M→C)			Step 3 (M→C→I)				Step 4 (M→C→I→O)				
	M	C	$\Delta m\%$	M	C	I	$\Delta m\%$	M	C	I	O	$\Delta m\%$
JT	1.71	0.00	-	1.03	1.21	0.00	-	5.15	1.38	1.80	2.16	-
FT	21.92	0.00	10.28	23.24	23.74	0.01	15.09	16.64	33.94	25.56	0.64	16.94
FE	11.85	**0.00**	5.16	15.06	4.86	**0.00**	5.96	19.62	11.54	20.77	2.16	11.22
LwF (Li and Hoiem 2017)	20.06	0.02	9.34	24.17	23.56	**0.00**	15.33	12.13	23.54	27.50	0.54	13.59
DyTox (Douillard et al. 2022)	9.24	1.00	4.33	21.11	3.25	0.70	7.68	19.93	4.40	24.14	12.14	12.90
L2P (Wang et al. 2022b)	8.49	0.15	3.52	16.38	4.13	0.84	6.44	17.36	11.59	15.98	0.92	9.10
S-iPr. (Wang, Huang, and Hong 2022)	**3.20**	**0.00**	**0.76**	6.20	**1.13**	2.09	2.41	11.35	**2.38**	2.09	3.01	2.18
Ours	6.52	**0.00**	2.45	**5.41**	1.57	**0.00**	**1.60**	**10.62**	2.54	**1.44**	**0.31**	**1.17**

Table 1: Comparison to other IL methods. Arrows indicate the order of learning. We measure both current performance and relative performance by HTER (%) and $\Delta m\%$, respectively. Lower $\Delta m\%$ indicates better overall performance.

Methods	Step 2 (S-P→O-R)			Step 2 (O-P→S-R)		
	S-P	O-R	$\Delta m\%$	O-P	S-R	$\Delta m\%$
JT	0.00	0.63	-	1.02	0.12	-
FT	31.17	2.77	16.66	28.20	1.78	14.56
FE	9.92	2.47	5.89	6.74	0.65	3.15
Ours	**0.65**	**0.63**	**0.33**	**6.65**	**0.28**	**2.92**

Table 2: Evaluation of cross-domain and cross-attack-type settings among SiW-Replay/SiW-Print (S-R/S-P) and OULU-Replay/OULU-Print (O-R/O-P) datasets.

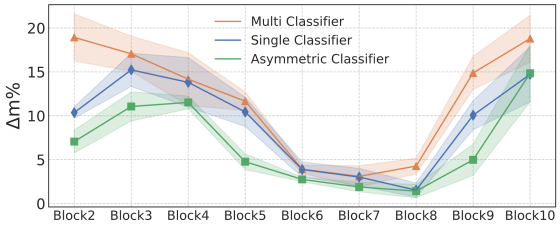

Figure 4: The results of different positions of IwR with different classifiers.

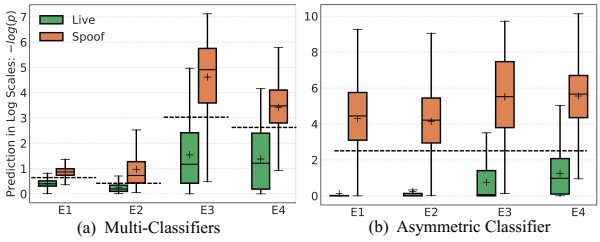

Figure 5: Visualization of PPD from Multi classifiers (a) and Asymmetric classifier (b).

and form a new dataset as SiW-Replay/SiW-Print. Similarly, we construct OULU-Replay/OULU-Print dataset. Then we conduct incremental experiments in cross-domain and cross-attack-type settings. For example, in the S-P→O-R setting, the model is trained on SiW-Print in step 1 and on OULU-Replay in step 2. As shown in Table 2, we can observe that the proposed method reduces the performance degradation by 16.33% and 11.64% in terms of S-P→O-R and O-P→S-R compared with FT, respectively. The results demonstrate the excellent anti-forgetting ability of our method.

Results of Cross-Domain in Incremental Settings. In this section, We conduct cross-dataset testing in common Leave-One-Out (LOO) settings of PAD domains. The comparison PAD methods include DA-based and DG-based methods. In contrast, IL settings have more stringent conditions that the model has to be trained on three datasets domain by domain, while DA-based and DG-based methods give the joint training performance, where the model is trained simultaneously on three datasets. It should be noted that sequential training will theoretically affect the ability of the model to learn a generalizable representation due to catastrophic forgetting. Nevertheless, as shown in Table 3, our approach outperforms IL-based methods and can achieve comparable performance with DA-based and DG-based methods in cross-domain settings, which demonstrates the effectiveness of alleviating catastrophic forgetting and preserving generalization capability.

Ablation Studies

Ablations of Different Positions of IwR. We insert the IwR module into the ViT network, which is divided into the first M blocks and the last $N - M$ blocks, named shared encoder f_s and expert decoder f_e, respectively. Here we aim to evaluate the influence of value M with different classifiers. Figure 4 illustrates the results in step 4. We can observe that the best performance is achieved by inserting IwR into the 8th block. The main reason is that the deeper shared encoder f_s brings more compact features within each domain for IwR, while deeper expert decoder f_e with ADE blocks provides a stronger potential for learning domain-specific knowledge separately and domain-invariant knowledge sequentially. Therefore, we trade off the performance of each module and set M to 8 for all experiments.

	Methods	O&C&I to M		O&M&I to C		O&C&M to I		I&C&M to O	
		HTER(%)	AUC(%)	HTER(%)	AUC(%)	HTER(%)	AUC(%)	HTER(%)	AUC(%)
DA	SDA (Wang et al. 2021a)	15.40	91.80	24.50	84.40	15.60	90.10	23.10	84.30
	VLAD (Wang et al. 2021b)	11.43	96.44	20.79	86.32	12.29	92.95	21.2	86.93
	GDA (Zhou et al. 2022b)	**9.20**	**98.0**	**12.20**	**93.00**	**10.00**	**96.00**	**14.40**	**92.60**
DG	MADDG (Shao et al. 2019)	17.69	88.06	24.50	84.51	22.19	84.99	27.98	80.02
	D2AM (Chen et al. 2021)	12.70	95.66	20.98	85.58	15.43	91.22	15.27	90.87
	FGHV (Liu et al. 2022)	9.17	96.92	12.47	93.47	16.29	88.79	13.58	93.55
	AMEL (Zhou et al. 2022a)	10.23	96.62	11.88	94.39	18.6	88.79	**11.31**	**93.96**
	SSDG-R (Jia et al. 2020)	7.38	97.17	10.44	95.94	11.71	96.63	15.61	91.54
	SSAN-R (Wang et al. 2022a)	**6.67**	**98.75**	**10.00**	**96.67**	**8.88**	**96.79**	13.72	93.63
IL	LwF (Li and Hoiem 2017)	17.14	98.18	33.94	69.56	20.29	91.24	19.03	88.93
	L2P (Wang et al. 2022b)	13.57	93.23	22.28	83.45	11.82	95.25	31.74	76.08
	Ours	**5.71**	**98.19**	**13.22**	**91.94**	**11.25**	**95.44**	**12.47**	**94.22**

Table 3: Comparison to SOTA PAD methods on four cross-dataset benchmarks in different learning settings: DA, DG, and IL. Note that IL methods train on three datasets sequentially, while DA and DG methods train on three datasets together.

Expert Branch	Multi classifiers		Asymmetric classifier	
	Aaware	Agnostic	Aware	Agnostic
E_1	9.68	24.70	**6.33**	**10.62**
E_2	20.21		**7.14**	
E_3	24.28		**10.82**	
E_4	25.07		**8.52**	

Table 4: Comparison to different classifiers in different settings in step 4. Results are reported in HTER(%).

Figure 6: The Grad-CAM visualizations of class activation map. The first row show the maps from FT method and the second row show from our method.

Ablations of the Different Classifiers. We compare the performance of different classifiers. Figure 4 shows that the asymmetric classifier has the smallest performance drop in various IwR position settings. Single-classifier strategies suffer from catastrophic forgetting, while multi-classifier has lower performance due to inconsistencies of PPD. We further conduct quantitative analysis between multi-classifiers and asymmetric classifier in Table 4. We train four datasets incrementally and test samples from the previous domain M in domain-aware/agnostic forms. Domain-aware form means that samples are forwarded via designated expert branch, while the domain-agnostic form means that samples are gated by the IwR mechanism. For multi-classifiers, individual expert branches with corresponding classifiers excel in domain-aware form but face notable performance decline when aggregating outputs due to PPD inconsistencies. In contrast, the asymmetric classifier keeps the class center θ_l shared across different expert branches, achieving superior performance in domain-agnostic form.

Visualization and Analysis

Visualization of Output Distribution. We conduct a statistical analysis of predicted scores from different classifiers. Figure 5 presents the boxplot of distributions based on predicted scores. We observe that in multi-classifiers method, the decision boundary of live and spoof samples varies greatly among different expert branch. A uniform threshold cannot achieve optimal results for all classifier outputs. For asymmetric classifier, the decision boundary is similar among different expert branch. A uniform threshold can maintain satisfactory results in different domains.

Visualization of Grad-CAM. As shown in Figure 6, we utilize the Grad-CAM (Zhou et al. 2016) to illustrate class activation maps. We randomly select some samples from dataset C in step 4 and compare the activation map learned by our method and fine-tuning method. The activation map from fine-tuning is shown in the first row. It has a serious problem of catastrophic forgetting (HTER: 1.38%→33.94%), which makes activation not look conspicuous for live samples and attention to wrong areas for spoof samples. In contrast, the second row shows that our method preserves activations in facial areas for live samples and highlights spoof cue areas of print attack and replay attack.

Conclusion

In this paper, we propose a novel MDIL framework of PAD to mitigate catastrophic forgetting in previous domains. Specifically, we design the ADE blocks equipped with learnable IwR to learn domain-specific knowledge separately without interference. Furthermore, an asymmetric classifier is designed to address the problem of PPD of live samples inconsistent in open appending domains. Extensive experiments with detailed analysis demonstrate the effectiveness.

References

Aljundi, R.; Babiloni, F.; Elhoseiny, M.; Rohrbach, M.; and Tuytelaars, T. 2018. Memory aware synapses: Learning what (not) to forget. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 139–154.

Atoum, Y.; Liu, Y.; Jourabloo, A.; and Liu, X. 2017. Face anti-spoofing using patch and depth-based CNNs. In *2017 IEEE International Joint Conference on Biometrics (IJCB)*, 319–328. IEEE.

Boulkenafet, Z.; Komulainen, J.; Li, L.; Feng, X.; and Hadid, A. 2017. OULU-NPU: A mobile face presentation attack database with real-world variations. In *2017 12th IEEE international conference on automatic face & gesture recognition (FG 2017)*, 612–618. IEEE.

Cai, R.; Cui, Y.; Li, Z.; Yu, Z.; Li, H.; Hu, Y.; and Kot, A. 2023. Rehearsal-Free Domain Continual Face Anti-Spoofing: Generalize More and Forget Less.

Chen, Z.; Yao, T.; Sheng, K.; Ding, S.; Tai, Y.; Li, J.; Huang, F.; and Jin, X. 2021. Generalizable representation learning for mixture domain face anti-spoofing. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, 1132–1139.

Chenshen, W.; HERRANZ, L.; Xialei, L.; et al. 2018. Memory replay GANs: Learning to generate images from new categories without forgetting [C]. In *The 32nd International Conference on Neural Information Processing Systems, Montréal, Canada*, 5966–5976.

Chingovska, I.; Anjos, A.; and Marcel, S. 2012. On the effectiveness of local binary patterns in face anti-spoofing. In *2012 BIOSIG-proceedings of the international conference of biometrics special interest group (BIOSIG)*, 1–7. IEEE.

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

Douillard, A.; Ramé, A.; Couairon, G.; and Cord, M. 2022. Dytox: Transformers for continual learning with dynamic token expansion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 9285–9295.

Freitas Pereira, T. d.; Anjos, A.; Martino, J. M. D.; and Marcel, S. 2012. LBP-TOP based countermeasure against face spoofing attacks. In *Asian Conference on Computer Vision*, 121–132. Springer.

Guo, X.; Liu, Y.; Jain, A.; and Liu, X. 2022. Multi-domain Learning for Updating Face Anti-spoofing Models. In *European Conference on Computer Vision*, 230–249. Springer.

Hou, S.; Pan, X.; Loy, C. C.; Wang, Z.; and Lin, D. 2019. Learning a unified classifier incrementally via rebalancing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 831–839.

Hu, C.; Zhang, K.-Y.; Yao, T.; Ding, S.; Li, J.; Huang, F.; and Ma, L. 2021. An End-to-end Efficient Framework for Remote Physiological Signal Sensing. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2378–2384.

Jia, M.; Tang, L.; Chen, B.-C.; Cardie, C.; Belongie, S.; Hariharan, B.; and Lim, S.-N. 2022. Visual prompt tuning. *arXiv preprint arXiv:2203.12119*.

Jia, Y.; Zhang, J.; Shan, S.; and Chen, X. 2020. Single-side domain generalization for face anti-spoofing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 8484–8493.

Jia, Y.; Zhang, J.; Shan, S.; and Chen, X. 2021. Unified unsupervised and semi-supervised domain adaptation network for cross-scenario face anti-spoofing. *Pattern Recognition*, 115: 107888.

Kanakis, M.; Bruggemann, D.; Saha, S.; Georgoulis, S.; Obukhov, A.; and Gool, L. V. 2020. Reparameterizing convolutions for incremental multi-task learning without task interference. In *European Conference on Computer Vision*, 689–707. Springer.

Kim, T.; Kim, Y.; Kim, I.; and Kim, D. 2019. Basn: Enriching feature representation using bipartite auxiliary supervisions for face anti-spoofing. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, 0–0.

Kirkpatrick, J.; Pascanu, R.; Rabinowitz, N.; Veness, J.; Desjardins, G.; Rusu, A. A.; Milan, K.; Quan, J.; Ramalho, T.; Grabska-Barwinska, A.; et al. 2017. Overcoming catastrophic forgetting in neural networks. *Proceedings of the national academy of sciences*, 114(13): 3521–3526.

Li, H.; Li, W.; Cao, H.; Wang, S.; Huang, F.; and Kot, A. C. 2018. Unsupervised domain adaptation for face anti-spoofing. *IEEE Transactions on Information Forensics and Security*, 13(7): 1794–1809.

Li, Z.; and Hoiem, D. 2017. Learning without forgetting. *IEEE transactions on pattern analysis and machine intelligence*, 40(12): 2935–2947.

Liu, S.; Lu, S.; Xu, H.; Yang, J.; Ding, S.; and Ma, L. 2022. Feature generation and hypothesis verification for reliable face anti-spoofing. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, 1782–1791.

Liu, S.; Zhang, K.-Y.; Yao, T.; Bi, M.; Ding, S.; Li, J.; Huang, F.; and Ma, L. 2021a. Adaptive normalized representation learning for generalizable face anti-spoofing. In *Proceedings of the 29th ACM International Conference on Multimedia*, 1469–1477.

Liu, S.; Zhang, K.-Y.; Yao, T.; Sheng, K.; Ding, S.; Tai, Y.; Li, J.; Xie, Y.; and Ma, L. 2021b. Dual reweighting domain generalization for face presentation attack detection. *arXiv preprint arXiv:2106.16128*.

Liu, Y.; Jourabloo, A.; and Liu, X. 2018. Learning deep models for face anti-spoofing: Binary or auxiliary supervision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 389–398.

Liu, Y.; Stehouwer, J.; Jourabloo, A.; and Liu, X. 2019. Deep tree learning for zero-shot face anti-spoofing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 4680–4689.

Mallya, A.; and Lazebnik, S. 2018. Packnet: Adding multiple tasks to a single network by iterative pruning. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 7765–7773.

Ostapenko, O.; Puscas, M.; Klein, T.; Jahnichen, P.; and Nabi, M. 2019. Learning to remember: A synaptic plasticity driven framework for continual learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 11321–11329.

Patel, K.; Han, H.; and Jain, A. K. 2016. Secure face unlock: Spoof detection on smartphones. *IEEE transactions on information forensics and security*, 11(10): 2268–2283.

Pérez-Cabo, D.; Jiménez-Cabello, D.; Costa-Pazo, A.; and López-Sastre, R. J. 2020. Learning to Learn Face-PAD: a lifelong learning approach. In *2020 IEEE International Joint Conference on Biometrics (IJCB)*, 1–9. IEEE.

Rebuffi, S.-A.; Bilen, H.; and Vedaldi, A. 2018. Efficient parametrization of multi-domain deep neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 8119–8127.

Rostami, M.; Spinoulas, L.; Hussein, M.; Mathai, J.; and Abd-Almageed, W. 2021. Detection and continual learning of novel face presentation attacks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 14851–14860.

Shao, R.; Lan, X.; Li, J.; and Yuen, P. C. 2019. Multi-adversarial discriminative deep domain generalization for face presentation attack detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 10023–10031.

Wang, G.; Han, H.; Shan, S.; and Chen, X. 2019. Improving cross-database face presentation attack detection via adversarial domain adaptation. In *2019 International Conference on Biometrics (ICB)*, 1–8. IEEE.

Wang, G.; Han, H.; Shan, S.; and Chen, X. 2020a. Unsupervised adversarial domain adaptation for cross-domain face presentation attack detection. *IEEE Transactions on Information Forensics and Security*, 16: 56–69.

Wang, J.; Zhang, J.; Bian, Y.; Cai, Y.; Wang, C.; and Pu, S. 2021a. Self-domain adaptation for face anti-spoofing. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, 2746–2754.

Wang, J.; Zhao, Z.; Jin, W.; Duan, X.; Lei, Z.; Huai, B.; Wu, Y.; and He, X. 2021b. VLAD-VSA: Cross-domain face presentation attack detection with vocabulary separation and adaptation. In *Proceedings of the 29th ACM International Conference on Multimedia*, 1497–1506.

Wang, Y.; Huang, Z.; and Hong, X. 2022. S-Prompts Learning with Pre-trained Transformers: An Occam's Razor for Domain Incremental Learning. *arXiv preprint arXiv:2207.12819*.

Wang, Z.; Wang, Z.; Yu, Z.; Deng, W.; Li, J.; Gao, T.; and Wang, Z. 2022a. Domain generalization via shuffled style assembly for face anti-spoofing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 4123–4133.

Wang, Z.; Yu, Z.; Zhao, C.; Zhu, X.; Qin, Y.; Zhou, Q.; Zhou, F.; and Lei, Z. 2020b. Deep spatial gradient and temporal depth learning for face anti-spoofing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 5042–5051.

Wang, Z.; Zhang, Z.; Lee, C.-Y.; Zhang, H.; Sun, R.; Ren, X.; Su, G.; Perot, V.; Dy, J.; and Pfister, T. 2022b. Learning to prompt for continual learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 139–149.

Wen, D.; Han, H.; and Jain, A. K. 2015. Face spoof detection with image distortion analysis. *IEEE Transactions on Information Forensics and Security*, 10(4): 746–761.

Wu, Y.; Chen, Y.; Wang, L.; Ye, Y.; Liu, Z.; Guo, Y.; and Fu, Y. 2019. Large scale incremental learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 374–382.

Yu, Z.; Li, X.; Shi, J.; Xia, Z.; and Zhao, G. 2021. Revisiting pixel-wise supervision for face anti-spoofing. *IEEE Transactions on Biometrics, Behavior, and Identity Science*, 3(3): 285–295.

Zenke, F.; Poole, B.; and Ganguli, S. 2017. Continual learning through synaptic intelligence. In *International Conference on Machine Learning*, 3987–3995. PMLR.

Zhang, K.-Y.; Yao, T.; Zhang, J.; Liu, S.; Yin, B.; Ding, S.; and Li, J. 2021. Structure destruction and content combination for face anti-spoofing. In *2021 IEEE International Joint Conference on Biometrics (IJCB)*, 1–6. IEEE.

Zhang, Z.; Yan, J.; Liu, S.; Lei, Z.; Yi, D.; and Li, S. Z. 2012. A face antispoofing database with diverse attacks. In *2012 5th IAPR international conference on Biometrics (ICB)*, 26–31. IEEE.

Zhou, B.; Khosla, A.; Lapedriza, A.; Oliva, A.; and Torralba, A. 2016. Learning deep features for discriminative localization. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2921–2929.

Zhou, K.; Liu, Z.; Qiao, Y.; Xiang, T.; and Loy, C. C. 2021. Domain generalization in vision: A survey. *arXiv preprint arXiv:2103.02503*.

Zhou, Q.; Zhang, K.-Y.; Yao, T.; Yi, R.; Ding, S.; and Ma, L. 2022a. Adaptive Mixture of Experts Learning for Generalizable Face Anti-Spoofing. *arXiv preprint arXiv:2207.09868*.

Zhou, Q.; Zhang, K.-Y.; Yao, T.; Yi, R.; Sheng, K.; Ding, S.; and Ma, L. 2022b. Generative domain adaptation for face anti-spoofing. *arXiv preprint arXiv:2207.10015*.