Pattern Recognition 179 (2026) 113640

Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/pr

Domain-generalizable face anti-spoofing with patch-based multi-tasking and artifact pattern conversion

Seungjin Jung [a,b,1], Yonghyun Jeong [b], Minha Kim [b,c,1], Jimin Min [b,d,1], Youngjoon Yoo [a,b], Jongwon Choi [a,e,f,*]

[a] Department of Artificial Intelligence, Chung-Ang University, Seoul, 06974, Korea
[b] Team of Image Vision, Naver Cloud, Seongnam, 13561, Korea
[c] Department of Human-centric AI, Nota AI, Seoul, 06164, Korea
[d] Department of Computer Engineering, Hanbat National University, Daejeon, 34158, Korea
[e] Department of Advanced Imaging, GSAIM, Chung-Ang University, Seoul, 06974, Korea
[f] Department of Metaverse Convergence, Chung-Ang University, Seoul, 06974, Korea

ARTICLE INFO

Keywords:
Domain generalizable face anti-spoofing
Pattern conversion GANs
Disentangle texture and contents
Patch based multi task learning

ABSTRACT

Face Anti-Spoofing (FAS) algorithms, designed to secure face recognition systems against spoofing, struggle with limited dataset diversity, impairing their ability to handle unseen visual domains and spoofing methods. We introduce the Pattern Conversion Generative Adversarial Network (PCGAN) to enhance domain generalizable in FAS. PCGAN effectively disentangles latent vectors for spoof artifacts and facial features, allowing to generate the images with diverse artifacts. We further incorporate patch-based and multi-task learning to tackle partial attacks and overfitting issues to facial features. Our extensive experiments validate PCGAN's effectiveness in domain generalization and detecting partial attacks, giving a substantial improvement in facial recognition security.

1. Introduction

Face recognition technology has made remarkable progress in recent years and is now widely deployed in various security systems. However, these recognition systems remain vulnerable to spoofing attacks in which adversaries deceive the system using either direct or indirect manipulation strategies [1,2]. Direct attacks, also known as presentation attacks, target the camera sensor by presenting physical artifacts such as printed photos or replayed videos. In contrast, indirect attacks manipulate data after acquisition, including digital attacks such as deepfake-based identity manipulation. In practice, attackers typically lack access beyond the input acquisition stage [1]. Therefore, Face Anti-Spoofing (FAS) algorithms have been developed to detect presentation attacks by distinguishing live face from spoofed face presented through physical media, such as printed photographs or electronic displays.

Although early FAS models have shown effectiveness in controlled scenarios, their performance degrades when exposed to unseen capturing environments (*e.g. brand new camera, irregular lighting conditions*) and new types of presentation attacks. The challenge lies in the lack of diversity in both subject identities and capture environments in existing training datasets, which hinders the generalization of FAS algorithms to real-world scenarios. Commonly used FAS datasets [3–6] typically contain fewer than 100 identities and lack diversity in capture environments, such as types of recapturing devices and illumination conditions.

Since it is infeasible to cover all possible scenarios involving diverse capturing environments and presentation attacks, researchers have improved the generality of detectors based on two major approaches: domain adaptation and domain generalization. Domain Adaptation for FAS (DAFAS) [7,8] incorporates domain adaptation techniques into face anti-spoofing by leveraging data from a specific source domain to progressively adapt the model to a distinct target domain. In contrast, Domain Generalizable FAS (DGFAS) [9–12] trains a FAS model to learn domain-invariant features using a multi-source domain dataset without relying on target domain data during training.

DAFAS achieves strong performance despite the absence of target-domain labels; however, it requires a dual-phase training procedure and access to target-domain data, which is often impractical in real-world scenarios. In contrast, DGFAS learns more generalized liveness and spoof features by leveraging multiple training domains, enabling competitive performance without target-domain data or additional model adaptation. Nevertheless, the limited diversity of existing FAS datasets still restricts generalization in unconstrained environments.

* Corresponding author.
E-mail address: choijw@cau.ac.kr (J. Choi).
[1] Authors Seungjin Jung, Jimin Min, and Minha Kim conducted this research during their internship at Naver Cloud.

https://doi.org/10.1016/j.patcog.2026.113640
Received 27 April 2025; Received in revised form 1 March 2026; Accepted 27 March 2026
Available online 30 March 2026

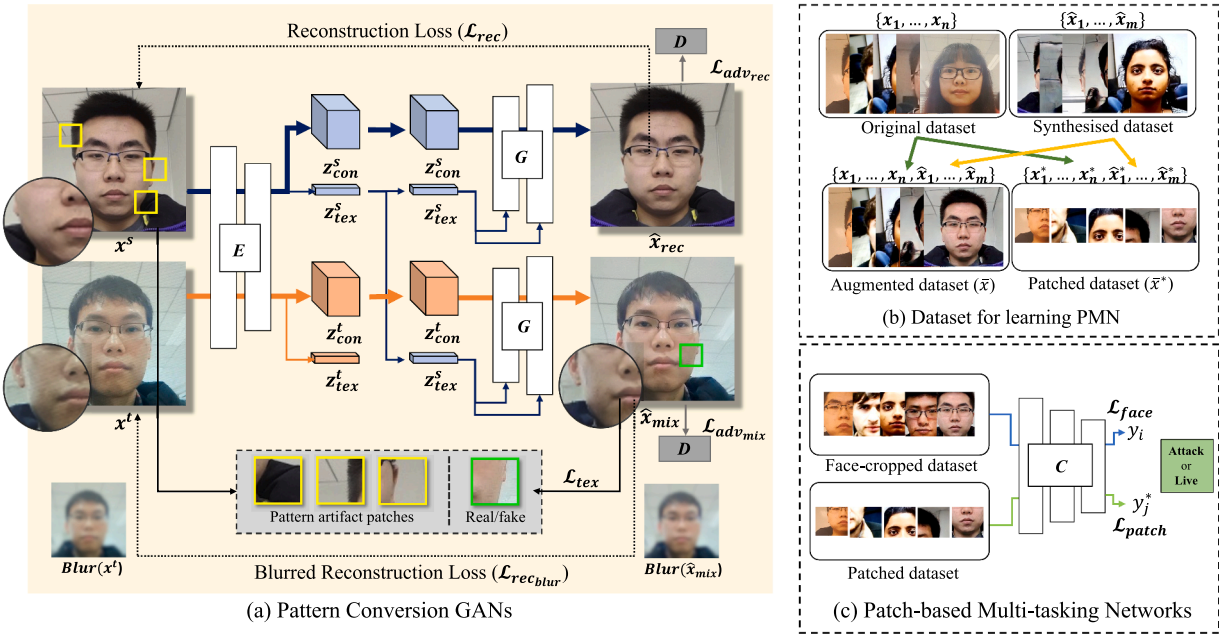

Fig. 1. Overall framework. (a) shows the disentanglement and conversion of artifact patterns by combining spatial content features and artifact representations from different images. (b) PCGAN-based data augmentation and construction of patched datasets for training. (c) Patch-based Multi-tasking Network (PMN), which jointly learns from patched and full face-cropped images.

To address this limitation, we propose the Pattern Conversion Generative Adversarial Network (PCGAN), which disentangles spoof artifacts and facial content to generate diverse synthetic FAS images with different artifact types. This strategy enriches training data diversity and enhances domain-generalizable feature learning. In addition, we incorporate patch-based learning and multi-task learning in the detector to handle partial spoofing and mitigate identity overfitting. Patch-based learning adopts a self-supervised scheme with randomly sampled artifact regions, while multi-task learning jointly exploits full-face images and synthesized patches during training.

The key contributions of our work are given as follows[1]:

- We propose Pattern Conversion Generative Adversarial Network (PCGAN) which effectively disentangles latent vectors corresponding to spoof artifacts and facial contents.
- By using disentangled features, we generate images that combine various spoof artifacts with a single facial identity, thereby enhancing the diversity of training data for Face Anti-spoofing (FAS).
- We propose patch-based and multi task learning methods to overcome partial spoofing attacks and the overfitting issue to well-aligned and limited face images.
- Through extensive experiments, we verify the effectiveness of our method not only for domain generalization but also for partial attack detection.

2. Related work

In this chapter, we discuss previous studies on DGFAS tasks, including data augmentation techniques using generative models for FAS and the patch-based FAS training approach. We present a comparison of our approach with these prior works.

2.1. Domain generalization for FAS

The field of Face Anti-Spoofing (FAS) has seen several notable studies addressing the challenges associated with dataset diversity and domain generalization. In this section, we discuss relevant works that have contributed to the advancement of FAS techniques. Huang [12] introduced global attention learning, while Wang [13] and Liao [14] proposed transformer-based learning methods for domain invariance. Wang et al. [15] utilized consistency regularization to perform domain generalization. They separated attack samples by domain and learned decision boundaries, while aggregating live samples across domains, resulting in a more generalizable feature space. Despite these advances, luminance and lightness variations across capture environments still cause substantial shifts in feature distributions, leading to inconsistent representations of spoof samples. To address this issue, we adopt an image-level approach that combines a generative model for learning spoofing artifact with a patch-based learning strategy focusing on localized image regions.

2.2. Generative model for FAS

The field of Face Anti-Spoofing (FAS) faces significant challenges due to limited dataset diversity and the need for robust domain generalization. Although several FAS datasets have been released [3–6], they do not cover the wide range of spoofing techniques used by attackers. Moreover, the rapid advancement of display technology further complicates the FAS landscape. Unlike face recognition, which utilizes large-scale datasets with thousands of identities, FAS datasets typically include fewer than 100 identities, resulting in models that struggle to generalize to new, unseen data. To address these issues, generative models have been employed to overcome dataset diversity limitations. Techniques such as StyleGAN [16], domain adaptation [17], and StyleAssemble [18] have been proposed to enhance generalization by generating data with domain-specific information. For style transfer, Yadav and Ross [19] used a cycle consistency approach, while [18] employed Adaptive Instance Normalization (AdaIN). Beyond style transfer, recent studies have explored generative modeling for FAS, where [20] learn intrinsic liveness characteristics via real-face generation and [21] leverage lightness-aware representations for image synthesis. In contrast, we propose a Pattern Conversion GAN that converts spoof artifact patterns between live and spoof images. While prior generative approaches mainly focus on domain style adaptation or one-directional artifact removal, our network explicitly disentangles and manipulates spoof artifacts. By separating facial content from artifact patterns and recombining them

[1] The source code and dataset will be available upon publication.

across samples, our method increases spoof data diversity at the artifact level, thereby improving robustness and generalization.

2.3. Patch-based learning for FAS

Face recognition tasks exhibit high accuracy, but face images are susceptible to various attack types. To address these vulnerabilities, [22–25] proposed a FAS method using local features and depth maps of face images. However, their approach involved extracting small patches from face images and training them separately with a depth-based CNN model, resulting in no parameter sharing between models. Chuang et al. [26] overcame this limitation with a multi-task meta-learning framework using a U-net-based face parsing module and depth estimator for spoof classification, facilitating parameter sharing.

Studies by PatchNet [27,28], and [29] demonstrated the benefits of incorporating patch-level inputs in training data. Srivatsan et al. [10] achieved success by employing CLIP[30], enhancing data diversity and enabling models to learn features specific to local regions targeted in spoof attacks. Another recent work utilizing a multi-modal approach, CFPL-FAS [31], addresses the problem of domain generalization for FAS through textual prompt learning conditioned on content and style features. Inspired by these studies, our paper introduces auxiliary supervision by training a FAS model using patches based on the CLIP model, recognizing that relevant information can be derived from both face images and external parts through patch-based learning.

3. Methods

In this chapter, we introduce two networks to perform domain generalization in Face Anti-Spoofing (FAS) tasks. The first network is Pattern Conversion Generative Adversarial Networks (PCGAN), which can convert existing artifacts in the attack image to the target image or remove existing artifacts. The second network is a Patch-based Multi-tasking Network (PMN) to robustly detect unseen presentation attacks.

3.1. Pattern conversion GANs

The PCGAN aims, in the context of presentation attacks, to disentangle the presentation artifacts from the image elements excluding the artifacts. Then, in the following step, we could transfer these disentangled artifacts onto live images or remove the artifacts from presentation images.

Spoofing Artifact: Spoofing artifacts are commonly understood as visual patterns introduced by the attack medium and the re-capturing process, rather than by genuine facial characteristics. Specifically, print attacks often exhibit printing-process artifacts such as halftone dot patterns [32], while replay attacks may produce moiré-like interference caused by the interaction between display pixel structures and camera sensor sampling grids [33].

3.1.1. Architecture

In this study, we propose a model that explicitly extracts spoofing artifacts from presentation attack images and exploits them in two complementary ways: removing artifacts from attack images and transferring them onto real images. Since spoofing artifacts do not naturally appear in genuine facial images, contrasting real and attack images enables reliable separation of artifact patterns from facial content. Based on this observation, our framework builds upon a swapping auto-encoder architecture [34], with a modified encoder that employs reduced downsampling to preserve fine-grained spoofing artifacts. The proposed model comprises four main components: an encoder, a generator, a discriminator, and a patch discriminator. The overall framework is illustrated in Fig. 1(a).

Encoder: The encoder E maps an input image $x \in \mathbb{R}^{3\times1024\times1024}$ into two disentangled latent representations: an artifact pattern representation $z_{pat} \in \mathbb{R}^{8\times512\times512}$ and a facial content representation $z_{con} \in \mathbb{R}^{8}$. Unlike conventional encoders that employ multiple downsampling layers, our encoder applies downsampling only once to better preserve fine-grained spoofing artifact patterns. Specifically, we follow the architecture of [34] but reduce the parameter $netE\_num\_downsampling\_sp$ from 4 to 1, as excessive downsampling significantly degrades spatial resolution and leads to the loss of critical artifact information.

Generator: The z_{pat} and z_{con} of different dimensions derived from the encoder are used as inputs to the generator. We feed-forward the content that contains spatial information to the generator to effectively reconstruct the input image. On the other hand, the latent vectors for artifact patterns are given to the generator as the input to every CNN block through adaptive instance normalization for the integration of two disentangled latent vectors [35].

Discriminator: In order to maximize the fidelity of the image generated by the generator, we use the discriminator proposed by [35]. We also include an additional network called patch discriminator to construct an artifact-aware pattern conversion model.

3.1.2. PCGAN losses

Pattern Conversion Loss: Pattern conversion loss is designed to disturb the recognition between fake converted artifact patterns and original artifact patterns when artifact patterns in an image are converted. The pattern patch discriminator is trained not to distinguish between the artifacts in the mixed $\hat{x}^{mix}$ determined by z^{src}_{pat} encoded from x^{src}. Therefore, the pattern conversion loss is defined as follows:

$$\mathcal{L}_{pat}(E,G,D_{patch}) = \mathop{\mathbb{E}}_{x^{src},x^{tgt}\sim\mathbf{X},x^{src}\neq x^{tgt}}\left[-\log\left(D_{patch}(\mathrm{crop}(G(z^{tgt}_{con},z^{src}_{pat})),\mathrm{crop}(x^{src}))\right)\right], \tag{1}$$

where crop() function defines the operation of the patch-based crop from the given image, and $(z^{tgt}_{con}, z^{tgt}_{pat}) = E(x^{tgt})$ and $(z^{src}_{con}, z^{src}_{pat}) = E(x^{src})$.

Reconstruction Loss: We construct a reconstruction loss to ensure that the network preserves all the information of the input image. The image reconstruction loss, which allows the network to reproduce the same image from the input image $x \sim \mathbf{X} \subset \mathbb{R}^{224\times224\times3}$, is expressed as follows.

$$\mathcal{L}_{rec} = \mathbb{E}_{x\sim\mathbf{X}}\left[||x - G(E(x))||_2\right]. \tag{2}$$

Conventional swapping auto-encoders preserve the overall facial structure but often introduce unintended variations in fine-grained components such as the eyes, nose, and mouth. To improve robustness to such local structural changes, we introduce a blurred reconstruction loss that preserves the semantic content of the mixed image x^{mix}. Specifically, we apply a downsampling-based blurring operation that suppresses spoofing artifact patterns while retaining semantic facial information. The input resolution is reduced from 1024×1024 to 512×512, preserving global facial structure while effectively removing high-frequency artifact components. Since only a $1/2$ downsampling is applied, major facial structures remain largely intact. Based on this property, we enforce that images sharing the same content representation z_{con} but different artifact representations z_{pat} converge to the same representation in the blurred space. The blurred reconstruction loss is defined as follows:

$$\mathcal{L}_{rec_{blur}} = \mathop{\mathbb{E}}_{\substack{x^{src},x^{tgt}\sim\mathbf{X},\\ x^{src}\neq x^{tgt}}}\left[||\mathrm{blur}(x^{tgt}) - \mathrm{blur}(G(z^{src}_{pat},z^{tgt}_{con}))||_2\right], \tag{3}$$

where blur(·) denotes a blurring operation implemented via down sampling.

Adversarial Loss: The purpose of the adversarial loss is to maintain the visual fidelity of the reconstructed images $\hat{x}^{rec}$ and mixed images $\hat{x}^{mix}$, as shown below:

$$\begin{aligned}\mathcal{L}_{adv_{rec}}(E,G,D) &= \mathop{\mathbb{E}}_{x\sim\mathbf{X}}\left[-\log(D(G(E(x^{src}))))\right],\\ \mathcal{L}_{adv_{mix}}(E,G,D) &= \mathop{\mathbb{E}}_{\substack{x^{src},x^{tgt}\sim\mathbf{X},\\ x^{src}\neq x^{tgt}}}\left[-\log(D(G(E(x^{src},x^{tgt}))))\right].\end{aligned} \tag{4}$$

This resolves the incongruity of the mixed image and generates a realistic image.

Table 1
Text prompt templates for live and attack faces.

Prompt No.	Liveness Face Prompts	Attack Face Prompts
No.1	This is an example of a real face	This is an example of a spoof face
No.2	This is a bonafide face	This is an example of an attack face
No.3	This is a real face	This is not a real face
No.4	This is how a real face looks like	This is how a spoof face looks like
No.5	a photo of a real face	a photo of a spoof face
No.6	This is not a spoof face	a printout shown to be a spoof face

Total Loss: In each iteration of the training phase, the model conducts mixture and reconstruction, simultaneously. The total loss is obtained by combining Eqs. (1), (2), (3), and (4), and is formulated as follows:

$$\mathcal{L}_{PCGAN} = \mathcal{L}_{rec} + \mathcal{L}_{rec_{blur}} + \mathcal{L}_{adv_{rec}} + \mathcal{L}_{adv_{mix}} + \mathcal{L}_{pat}. \tag{5}$$

3.1.3. Artifact pattern conversion

To generate training samples for PMN, we augment the data using ground-truth labels. Given attack images $\{x^a\}_{i=1}^n$, a synthetic attack image $\hat{x}^{\text{atk}}$ is produced by transferring spoofing artifact patterns from an attack image x_h^a onto a live image x_g^{liv}, where $g, h \in [1, \dots, n]$. Conversely, a synthetic live image $\hat{x}^{\text{liv}}$ is obtained by removing spoofing artifacts from an attack image and replacing them with live facial content from x^{liv}. This symmetric conversion enables balanced augmentation for both classes while preserving the original class ratio.

3.2. Patch-based multi-tasking network

Patch-based Multi-tasking Network (PMN) consists of CLIP [30] C, Multi Layer Perceptron (MLP) F, and Fully Connected Layer (FC) M, and we train PMN by the label $y \sim \mathbf{Y} \subset \mathbb{R}^2$, the description text $t \sim \mathbf{T}$, and the combined image $\bar{x} \sim \bar{\mathbf{X}}$ which is a combination of the generated image $\hat{x} \sim \hat{\mathbf{X}}$ and the original image $x \sim \mathbf{X}$. All images are resized to 224×224.

3.2.1. Overall framework

The use of patch-based models, as demonstrated in previous studies [22,27], can increase the diversity of data and encourage the network to learn features specific to presentation attacks in localized areas. We use patched images $\bar{x}^* \sim \bar{\mathbf{X}}^*$ obtained by randomly cropping images including parts of the face from the whole image $\bar{x}$. The methods for constructing patched images and augmented images for PMN training are shown in Fig. 1(b).

Multi-Task Learning (MTL) aims to share knowledge while simultaneously learning data with similar but different tasks. We employ the CLIP [30] with an MLP to share knowledge in MTL, learning $\bar{x}^*$ and $\bar{x}$ simultaneously. Therefore, PMN not only learns presentation attack characteristics in full-face images but also learns spatial consistency between background and foreground regions. The last fully connected layer of PMN separates each of the two tasks. After the training, only the face-cropped image is used during inference. The overall framework is shown in Fig. 1(c).

3.2.2. PMN losses

CLIP Loss: Since previous work [10] shows that CLIP [30] is effective for the Presentation Attack Detection task, we use CLIP as the backbone network. CLIP is trained on a large set of image and text pairs taken from the Internet and is fine-tuned to suit the task when used. CLIP consists of image encoder C_I and text encoder C_T, where description texts $t \sim T$ for each label similar to [10]. The description texts consist of six sentences [10] each for attack and live depicted as in Table 1, and they are transformed into text embedding features through a text encoder C_T. We calculate the average for each attack and live feature embedding and the cosine similarity between the mean text and image embedding features. Thus, we can design CLIP loss as follows:

$$\mathcal{L}_{clip} = \mathop{\mathbb{E}}_{(\bar{x},t,y)\sim\bar{\mathbf{X}},\mathbf{T},\mathbf{Y}} \Big[CE\big(\langle C_I(\bar{x}), \mathop{\mathbb{E}}_{t_a,t_l\sim T}[C_T(t)]\rangle, y\big)\Big], \tag{6}$$

where CE is a cross-entropy loss, $\langle,\rangle$ represents an inner product, and t_a and t_l mean text description elements for attack and live, respectively.

Multi-tasking Loss: During the training of PMN, we utilize both the full image, denoted as x, and its corresponding patch image that is x^* and obtained by cropping from the full image x as a random crop size scale $0.2 - 1.0$. Patch-level supervision using randomly cropped images x^* enhances robustness to partial attacks by enabling localized spoofing artifacts to directly contribute to the detection decision. Then, we apply the MLP approach to extract image features from the embedded image feature of CLIP [10,36]. Thus, we formulate PMN Loss defined as:

$$\mathcal{L}_{face} = \mathop{\mathbb{E}}_{(\bar{x},y)\sim(\bar{\mathbf{X}},\mathbf{Y})} \Big[CE\big(M_1(F(C(\bar{x}))), y\big)\Big], \tag{7}$$

$$\mathcal{L}_{patch} = \mathop{\mathbb{E}}_{(\bar{x}^*,y)\sim(\bar{\mathbf{X}}^*,\mathbf{Y})} \Big[CE\big(M_2(F(C(\bar{x}^*))), y\big)\Big]. \tag{8}$$

Center Loss: Center loss [37] is adopted to optimize the intermediate feature distribution, aiming for improved generalization capabilities for unknown attacks. It includes the distance from the sample to the center of the sample's class. c_{y_i} denotes the y_ith class center of features from face-cropped images. The center loss is calculated for each class (*i.e.,* live and attack) regardless of the domain.

$$\mathcal{L}_{center} = \mathop{\mathbb{E}}_{(\bar{x},y)\sim(\bar{\mathbf{X}},\mathbf{Y})} \left[\frac{1}{2} \left\| F(C(\bar{x})) - c_y \right\|_2^2 \right]. \tag{9}$$

Total Loss: Finally, our total loss for PMN can be written using Eqs. (7)–(9), as follows:

$$\mathcal{L}_{PMN} = \mathcal{L}_{clip} + \mathcal{L}_{face} + \mathcal{L}_{patch} + \alpha\mathcal{L}_{center} + \beta l_2, \tag{10}$$

where l_2 is the l_2-regularization, and α and β are the user-defined hyperparameters.

4. Experiments

4.1. Experimental setup

Dataset: We evaluate the five benchmark datasets including CASIA-FASD [3] (denoted as C), Idiap REPLAY-ATTACK [4] (denoted as I), OULU-NPU [6] (denoted as O), MSU-MFSD [5] (denoted as M), Rose Youtu [38], CASIA-SURF CeFA [39], CASIA-SURF [40], and WMCA [41]. For all the tables in the rest of the manuscript, the datasets including OULU [6], MSU-MFSD [5], Idiap Replay-attack [4], and CASIA-FASD [3] are abbreviated as $\{O, M, I, C\}$ respectively. Similarly, CASIA-SURF CeFA, CASIA-SURF, and WMCA are abbreviated as $\{C, S, W\}$.

Metric: To evaluate performance, we use various metrics commonly employed in FAS, i.e., Attack Presentation Classification Error Rate (APCER), Bona Fide Presentation Classification Error Rate (BPCER), and Average Classification Error Rate (ACER) to ensure fairness in comparison. For cross-domain testing, we adopt ACER and Area Under Curve (AUC) as evaluation metrics.

Detailed Experimental Configuration: Our experiments are conducted on an Nvidia RTX A6000 GPU with a batch size of 1, using the Adam optimizer for 4000 iterations. The initial learning rate is 1e-6 with betas 0.9, 0.999. For face region cropping, we use dataset-provided face location information or MTCNN [46], and apply a padding value of 0.6. The hyperparameters α and β are set to 0.2 and 1e-6, respectively. Testing is performed using face-cropped images only.

4.2. Quantitative comparison

This section presents the quantitative results of our method compared to various approaches across different cross-domain settings. First, we evaluate models with multiple training source domains in Table 2.

Table 2
Best-epoch comparison on the DG-FAS benchmark. ACER and AUC are reported under four DG protocols (O, M, I, C denote OULU, MSU-MFSD, Replay-Attack, and CASIA-FASD). Bold and underlined values indicate the best and second-best performance, respectively. * denotes the use of CelebA-Spoof.

Method	OCI→M		OMI→C		OCM→I		ICM→O		Average	
	ACER (%)	AUC (%)	ACER (%)	AUC (%)	ACER (%)	AUC (%)	ACER (%)	AUC (%)	ACER (%)	AUC (%)
MA-Net [42]	20.80	–	25.60	–	24.70	–	26.30	–	24.35	–
SSAN-R [18]	6.67	98.75	10.00	96.67	8.88	96.79	13.72	93.63	9.82	96.46
AFD [12]	12.92	93.29	17.78	88.10	18.75	91.92	15.90	90.54	16.34	90.96
PatchNet [27]	7.10	98.46	11.33	94.58	13.40	95.67	11.82	95.07	10.90	95.95
DFDN [11]	5.20	98.39	8.00	97.45	7.71	95.56	11.01	95.22	7.98	96.66
SAFAS [9]	5.95	96.55	8.78	95.37	6.58	97.54	10.00	96.23	7.83	96.42
CA-FAS [43]	7.14	97.42	11.68	94.55	13.86	93.67	11.67	94.53	11.09	95.04
ViT&FA&CS [32]	4.62	98.92	7.28	97.02	10.89	97.05	6.77	98.25	7.39	97.81
AG-FAS [20]	5.71	98.03	5.44	98.55	6.71	98.23	9.43	96.62	6.82	97.86
CA-MoEiT [44]	2.88	98.76	7.89	97.70	6.18	98.94	9.72	96.22	6.67	97.91
FSFM ViT-B [33]	3.78	99.15	3.16	99.41	4.63	99.03	7.68	97.11	4.81	98.68
CCPE [45]	3.10	99.21	1.33	99.36	6.08	94.36	5.57	98.49	4.02	97.86
CFPL [31]	3.09	**99.45**	2.56	99.10	5.43	98.41	3.33	99.05	3.60	99.00
FLIP-MCL* [10]	5.00	98.35	**0.54**	**99.98**	4.25	99.07	3.70	99.28	3.37	99.17
CFPL* [31]	**1.43**	99.28	2.56	99.10	5.43	98.41	**2.50**	**99.42**	2.98	99.05
PCGAN only(Ours)	7.58	96.94	2.22	99.72	4.88	99.22	4.69	99.10	4.84	98.75
Ours	2.50	99.35	2.04	99.32	**3.33**	**99.11**	3.29	99.08	**2.79**	**99.22**

Table 3
Domain generalization results on the FAS benchmark. ACER and AUC are averaged over the last 10 epochs. Parentheses indicate ACER differences from Table 2 as a measure of learning stability. The same four protocols as Table 2 are used.

Method	OCI→M		OMI→C		OCM→I		ICM→O		Average	
	ACER (%)	AUC (%)	ACER (%)	AUC (%)	ACER (%)	AUC (%)	ACER (%)	AUC (%)	ACER (%)	AUC (%)
SSAN-R [18]	21.79(15.12)	84.06	26.44(16.44)	78.44	35.39(26.51)	70.13	25.72(12.00)	79.37	27.34(17.52)	78.00
PatchNet [27]	25.92(18.82)	83.43	36.26(24.93)	71.38	29.75(16.35)	80.53	23.49(11.67)	84.62	28.86(17.96)	79.99
SAFAS [9]	14.36(8.41)	92.06	19.40(10.62)	88.69	11.48(4.90)	95.74	11.29(1.29)	95.23	14.13(6.26)	92.93
FLIP-MCL [10]	15.00(10.00)	93.37	**3.22**(2.68)	**99.44**	10.25(6.00)	96.56	5.75(2.05)	**98.48**	8.56(5.19)	96.96
Ours	**8.21**(5.71)	**96.95**	5.41(3.37)	98.76	**9.55**(6.22)	**96.72**	**5.51**(2.22)	98.23	**7.17**(4.38)	**97.67**

Second, following [9], we report the average performance over the last 10 epochs and evaluate learning stability by measuring the gap between the best epoch and the averaged results, as shown in parentheses in Table 3. Third, we extend our evaluation to large-scale cross-domain and cross-ethnicity benchmarks, including CASIA-SURF CeFA, CASIA-SURF, and WMCA (CSW benchmarks). Compared to conventional DG-FAS protocols based on small-scale RGB datasets, these benchmarks introduce substantially higher intra-class variation and domain shifts. The results in Table 4 show that our method maintains strong performance under these challenging settings, confirming its scalability and robustness in more realistic scenarios.

4.2.1. Cross-domain testing

Table 2 reports ACER and AUC results of various methods under four cross-domain settings. Our method achieves the lowest ACER of 3.33% on OCM→I, and the second-best performance on the remaining three cases, where the best results are obtained by methods using CelebA-Spoof as extra data. Notably, among methods without CelebA-Spoof, our approach achieves the best ACER across all four cases. Moreover, considering both average ACER and AUC, our method outperforms all compared approaches, demonstrating state-of-the-art performance among CLIP-based methods.

4.2.2. Cross-domain testing of the average results

In cross-domain FAS, performance can vary significantly across epochs since the target domain is unseen during training. Following [9], we report the average ACER and AUC over the last 10 epochs to assess generalization. Table 3 compares our method with others under this protocol. For average ACER, our method achieves state-of-the-art performance in all cases except OMI→C. Notably, it records 8.21% on OCI→M, outperforming the second-best result by a large margin, and achieves the lowest error rates on OCM→I (9.55%) and ICM→O (5.51%). On OMI→C, our method attains the second-best result (5.41%), following FLIP [10]. Across all protocols, only our method consistently achieves ACER below 10%. For average AUC, our method attains the best performance on OMI→C and ICM→O. Overall, considering both ACER and AUC averages, our approach significantly outperforms competing methods, demonstrating strong and stable generalization.

4.2.3. Learning stability

Table 2 shows the best results, and Table 3 displays the average results for the last 10 epochs. In the FAS task, the model is challenging to converge, so the average of the last 10 epochs is considered an approximate convergence value. The difference in ACER values between Table 2 and Table 3, as indicated in parentheses on Table 3, represents how well the model converges close to the best results. For overall methods, while learning stability is not good in the case of OMI→C and OCM→I, learning stability is good in the OCI→M and ICM→O. The averages of OCI→M and ICM→O datasets are more stable than OMI→C and OCM→I datasets for learning. Our methods achieve the best stability result for one case and the second-best result for two cases. In terms of the average, our method attains the highest learning stability score.

4.3. Large-scale cross-domain evaluation on CSW benchmarks

Table 4 presents the ACER and AUC performance of various methods on the CSW large-scale benchmarks under three cross-domain settings: CS→W, SW→C, and CW→S. Compared with conventional small-scale RGB datasets, these benchmarks introduce larger subject diversity and more severe domain shifts, making them substantially more challenging. In terms of ACER, our method achieves the best average performance across the three protocols, with an average ACER of 10.78%,

Table 4
Results on CSW benchmarks. Evaluation metrics are ACER and AUC under cross-domain settings on CASIA-Surf-CeFA (C), CASIA-Surf (S), and WMCA (W), where bold and underlined values indicate the best and second-best performance, respectively.

Method	CS→W		SW→C		CW→S		Average	
	ACER (%)	AUC (%)	ACER (%)	AUC (%)	ACER (%)	AUC (%)	ACER (%)	AUC (%)
ViT	21.04	89.12	17.12	89.05	17.16	90.25	18.44	89.47
CLIP-V [30]	20.00	87.72	17.67	89.67	8.32	97.23	15.33	91.54
CLIP [30]	17.05	89.37	15.22	91.99	9.34	96.62	13.87	92.66
CA-MoEiT [44]	16.67	91.02	16.42	93.17	12.57	93.76	15.22	92.59
CoOp [47]	**9.52**	90.49	18.30	87.47	11.37	95.46	13.06	91.14
CFPL [45]	9.57	**94.25**	14.89	91.56	8.16	96.78	10.87	94.20
Ours	12.58	94.10	**11.88**	**94.96**	**7.89**	**97.67**	**10.78**	**95.58**

Table 5
For the ablation study, we report results on a domain-generalization FAS benchmark using ACER. A checkmark indicates the inclusion of each component in PMN and PCGAN, where CL, PL, and Syn denote Center Loss, Patch Loss, and synthesized images, respectively.

(a) Protocol 1: Best epochs

PMN		PCGAN	OCI→M	OMI→C	OCM→I	ICM→O	Avg.
CL	PL	Syn					
			10.42	4.44	7.75	6.11	7.18
	✓		10.00	2.04	6.67	3.38	5.52
✓			7.50	2.22	5.25	5.14	5.03
✓	✓		5.00	3.33	6.67	4.81	4.95
		✓	7.58	2.22	4.88	4.69	4.84
	✓	✓	7.92	2.22	6.83	**2.59**	4.89
✓		✓	5.42	2.41	6.12	4.48	4.61
✓	✓	✓	**2.50**	**2.04**	**3.33**	3.29	**2.79**

outperforming all compared approaches. Notably, our method obtains the lowest error rate of 7.89% in the CW→S setting and demonstrates competitive performance in the other two protocols. Regarding AUC, our method consistently achieves the highest average score 95.58%, indicating more reliable discrimination capability under large-scale cross-domain conditions. These results demonstrate that the proposed PCGAN and PMN framework generalizes effectively beyond traditional DG-FAS benchmarks and maintains strong robustness under more realistic and diverse evaluation scenarios.

4.4. Quantitative analysis

4.4.1. Effectiveness of components

This paragraph analyzes the contributions of three components in our framework: Center Loss (CL), Patch-based Learning (PL) in PMN, and synthesized images (Syn) generated by PCGAN. As shown in Table 5, the baseline without these components performs worst, confirming the need for additional supervision. Introducing PL consistently improves performance, demonstrating the effectiveness of patch-level supervision for localized spoof artifacts. Incorporating Syn further enhances cross-domain generalization, especially in challenging protocols such as OCM→I and ICM→O, while CL stabilizes feature distributions and improves robustness. The full model combining CL, PL, and Syn achieves the best overall performance, highlighting strong complementarity among the components: PL enables localized artifact reasoning, Syn increases domain diversity, and CL promotes feature compactness, resulting in robust and stable generalization.

4.4.2. Effectiveness of PCGAN-generated synthetic samples

To evaluate PCGAN under a controlled setting, we conduct an ablation study where the detector is trained without any real live images, using only original spoof images and PCGAN-generated synthetic samples. This design isolates the effect of artifact removal and injection from real live data. As shown in Table 6, the baseline achieves an average ACER of 4.95%. Replacing real live images with synthetic-live samples generated via Spoof→Live yields comparable performance (5.70% ACER), despite the absence of real live data, indicating that PCGAN effectively removes spoof artifacts while preserving semantic facial content. In contrast, training with synthetic-spoof samples from Live→Spoof results in a larger degradation (7.00% ACER), suggesting that artifact injection alone is insufficient to model real spoof diversity. Using both synthetic-live and synthetic-spoof samples improves performance (6.28% ACER) but remains inferior to the Spoof→Live setting, highlighting the asymmetric difficulty between artifact removal and injection. Overall, these results demonstrate that PCGAN learns controllable artifact-specific representations, with Spoof→Live enabling effective cross-domain training even without real live data.

Table 6
Ablation study on synthetic-live and synthetic-spoof samples generated by PCGAN. Spoof→Live replaces real live data with artifact-removed spoof images, while Live→Spoof replaces real spoof data with artifact-injected live images. ACER (%) is reported under four cross-domain protocols.

Spoof→Live	Live→Spoof	OCI→M	OMI→C	OCM→I	ICM→O	Average
		5.00	3.33	6.67	4.81	4.95
✓		5.42	4.44	8.75	4.17	5.70
	✓	7.50	6.67	10.00	3.82	7.00
✓	✓	7.50	3.33	11.25	3.02	6.28

Table 7
ACER (%) performance on different datasets. We train the model using either the original images (Original OULU) or synthesized images (Syn OULU).

(a)Protocol 1: Best epochs

Origin	Syn	OULU	MSU	Idiap	CASIA	Average
✓		**0.00**	**5.00**	7.83	4.63	4.37
	✓	8.33	25.00	16.33	7.96	14.41
✓	✓	**0.00**	7.50	**5.17**	**4.44**	**4.28**

Table 8
Experimental results show the impact of changing the random cropping size. The metrics used in this experiment are the ACER.

Range	OCI→M	OMI→C	OCM→I	ICM→O	Avg.
0.2-1.0	**2.50**	2.04	**3.33**	3.29	**2.79**
0.2-0.2	5.00	**1.11**	8.50	3.38	4.50
0.4-0.4	5.42	2.22	11.67	**2.22**	5.38
0.6-0.6	7.08	2.22	10.00	3.38	5.67
0.8-0.8	5.00	**1.11**	10.00	3.33	4.86

4.4.3. Generalization through PCGAN-based augmentation

We set up scenarios in which we train the model on the original image, synthesized image, and both, respectively, for the OULU

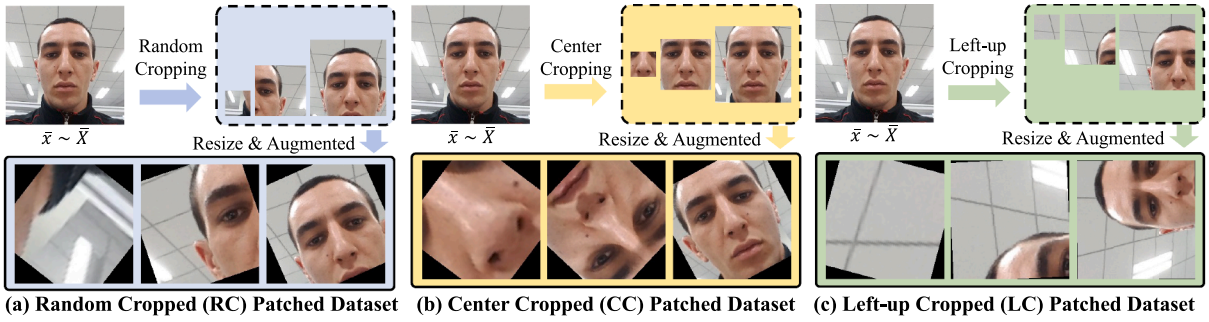

Fig. 2. Examples displayed each cropping approach.

Table 9
Experimental results on the domain-generalization for FAS benchmark. The metrics used in this experiment are the ACER. RC, CC, and LC mean Random Cropping, Center Cropping, and Left Up Cropping, respectively.

(a) Protocol 1: Best epochs

Method	OCI→M	OMI→C	OCM→I	ICM→O	Avg.
RC	**2.50**	2.04	**3.33**	3.29	**2.79**
CC	7.50	**1.11**	8.33	4.77	5.43
LC	4.17	2.22	10.00	**2.59**	4.74

dataset [6]. Subsequently, we validate each model on the test datasets for all datasets. Table 7 shows the results of the above scenario. Given training on the original dataset as the baseline, training on the synthesized dataset decreases performance compared to the baseline, while training on both improves the model's performance. Synthesized datasets show good synergy with original datasets, although they are not useful on their own. Therefore, images synthesized from PCGAN are effective enough as augmented datasets.

4.4.4. *Effectiveness of cropping size*

In this section, we analyze the effect of face cropping size on patch-based learning. As shown in Table 8, we compare random cropping with a scale range of 0.2-1.0 against fixed cropping scales (0.2, 0.4, 0.6, 0.8). Fixed cropping achieves comparable or better performance on OMI→C and ICM→O, but degrades on OCI→M and OCM→I, with particularly severe drops on OCM→I. In contrast, random cropping shows stable performance across all datasets. Although specific fixed scales perform best in certain environments, these results suggest that optimal crop sizes are dataset-dependent, motivating the use of random cropping to cover diverse scales.

4.4.5. *Various cropping approaches*

In this section, we analyze the impact of different cropping strategies on our method. To learn localized facial features (e.g., eyes and nose), we construct a patched dataset using various cropping approaches and conduct comparative experiments.

Random Cropping (RC) RC randomly selects cropping regions (Fig. 2(a)) with scales ranging from 0.2 to 1.0 of the image size, enabling the model to learn multi-scale localized features from diverse spatial locations.

Center Cropping (CC) CC selects cropping regions centered in the image (Fig. 2(b)) with scales ranging from 0.2 to 1.0, encouraging the model to learn multi-scale localized features around central facial regions.

Left-up Cropping (LC) LC selects cropping regions from the upper-left area of the image (Fig. 2(c)) with scales ranging from 0.2 to 1.0, promoting the learning of multi-scale localized features biased toward the upper-left facial region.

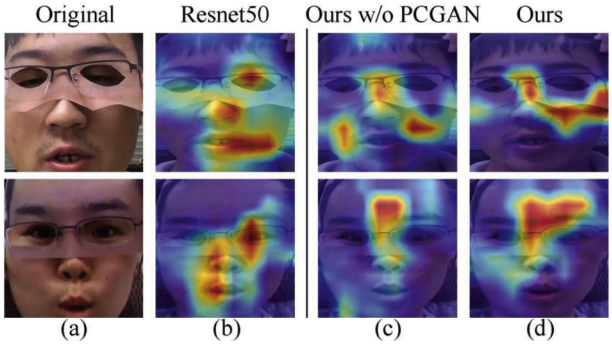

Fig. 3. Grad-CAM visualizations under the MOI→C protocol. Activation maps for the attack class on the ROSE-Youtu dataset are shown to evaluate robustness to unseen partial attacks. (a) Attack image. (b) ResNet50 without PMN and PCGAN. (c) Our model with PMN. (d) Our model with both PMN and PCGAN.

Quantitative Results Table 9 reports ACER results for different cropping strategies under four cross-domain settings. RC achieves the best performance on OCI→M and OCM→I, and the second-best on OMI→C and ICM→O, while CC and LC perform best on OMI→C and ICM→O, respectively. Overall, RC shows the most consistent performance across environments, indicating stronger generalization.

4.5. *Qualitative analysis*

Grad-CAM Visualization: Fig. 3 illustrates the grad-cam visualization to show the effect of the patch-based learning and synthesized data. Ours w/ patch, Ours w/ patch&syn, and ResNet w/o patch denote the proposed method employing patch-based learning, patch-based learning incorporated with synthetic samples, and baseline. For case (a-b), ResNet w/o patch falsely classifies the spoof images, partial attack case, to live. Furthermore, even for the correct classification, as shown in (c-d), ResNet w/o patch highlights the region unnecessary for the decision. Conversely, when applying our proposed method, without using the synthesized samples for training, the grad-cam directs semantically plausible regions to classify the spoof image. Also, the grad-cam results in row (d) show that the synthetic samples help the network focus more on spoofed regions. The grad-cam visualizations provide substantial clues why our proposed method performs robust classification for complex presentation cases such as partial attacks.

Conversion results from Artifact Patterns: This chapter visualizes PCGANs disentanglement of artifact patterns from spoof images. Fig. 4 shows the result of adding the artifact patterns of the attack image to the live image or separating and removing the artifact patterns from the attack image. artifact patterns in the example are moiré artifacts caused by the display and have linear characteristics. Therefore, after detecting edge components through Canny Detection, linear components can be visualized through Hough Transform. As shown in the results, it can be

Table 10
Computational cost comparison in terms of parameters and FLOPs. The generator is used only during training, so inference cost reflects the detector alone.

Method	Generator			Detector				Inference		
				Image Encoder		Text Encoder				
	Type	Parameters	Flops	Parameters	Flops	Parameters	Flops	Parameters	Flops	Inference Time
ViT [48]	None	–	–	86.19M	17.58G	–	–	86.19M	17.58G	0.007
FLIP-IT [10]	None	–	–	86.19M	17.58G	63.11M	35.81M	86.19M	17.58G	0.0010
FLIP-MCL [10]	None	–	–	86.19M	52.74G	83.05M	35.86M	86.19M	17.58G	0.0010
Ca-MoEiT [44]	None	–	–	86.19M	17.58G	–	–	86.19M	17.58G	–
AG-FAS [20]	Diffusion	1.07B	16.94T	86.19M	17.58G	–	–	86.19M	17.58G	–
Ours	GAN	109.03M	95G	86.19M	35.16G	63.11M	35.81M	86.19M	17.58G	0.0010

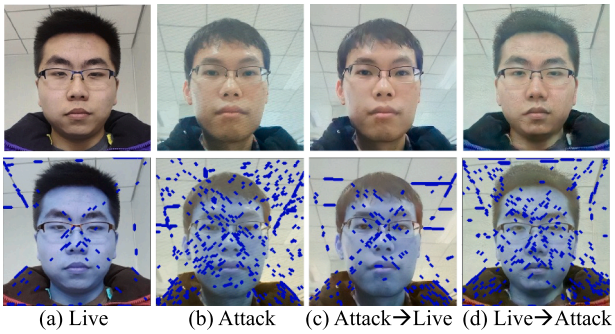

Fig. 4. Cross class results with Pattern Conversion GANs. The images show the artifact pattern detection results by Canny detection and Hough Transform. (c) shows the artifact-pattern removal results from (b). (d) inserts artifact patterns from (a) into (b).

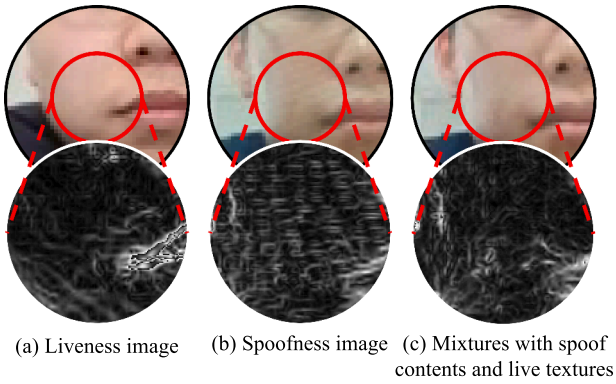

Fig. 5. Distinct visual artifacts extracted via Sobel filtering. (a)-(c) are taken from Fig. 1.

seen that the artifacts detected in the spoofed image are reduced, and the artifacts are injected into the real image.

Fig. 5 demonstrates the example of spoof textures exhibiting recurrent simple patterns originating from the Moiré effect. To emphasize the patterns on the second row, we obtained the gradient magnitude map using 3×3 Sobel filters in the x and y directions. In Fig. 5, (a) does not reveal the recurrent pattern, (b) clearly shows the pattern, and (c) demonstrates that the pattern has been removed by PCGANs.

4.6. Computational cost analysis

Table 10 compares the computational complexity of our method with recent state-of-the-art approaches. We report parameters and FLOPs for the generator (if applicable), detector, and inference stage. Although our framework includes a GAN-based generator (109.03M parameters, 95G FLOPs), it is used only during training and excluded at inference. Thus, the runtime cost equals that of the CLIP-based detector backbone (86.19M parameters, 17.58G FLOPs). Compared to diffusion-based methods such as AG-FAS, which require substantially higher generation cost, our approach significantly reduces training-time complexity while maintaining identical inference cost.

5. Conclusion

In this paper, we proposed a domain-generalizable face anti-spoofing framework that integrates a Pattern Conversion GAN (PCGAN) with a Patch-based Multi-tasking Network (PMN). By disentangling spoofing artifacts from facial content and recombining them across samples, the proposed method alleviates the limited diversity of existing FAS datasets and improves robustness to unseen domains and partial attacks. A key strength of our approach is its artifact-centric design. Unlike prior generative methods that focus on global appearance or domain styles, PCGAN explicitly models spoofing artifacts introduced by attack media and recapturing processes, enabling controllable artifact removal and injection. In addition, patch-based multi-task learning encourages the detector to jointly exploit global facial cues and localized evidence, leading to improved generalization in challenging cross-domain scenarios. Despite these advantages, the proposed method has limitations. The diversity of synthesized samples is bounded by the artifacts present in the training data, and the generator is intended for data augmentation rather than large-scale or real-time synthesis. Moreover, although extensive experiments demonstrate strong generalization, further evaluation on emerging attack types and more unconstrained real-world scenarios remains an important direction for future work. Nevertheless, this work provides a practical framework for improving domain generalization in face anti-spoofing. The proposed artifact disentanglement and patch-based learning strategies are readily extensible to other FAS backbones and potentially to broader biometric anti-spoofing tasks. Future work will explore richer artifact synthesis with advanced generative models, finer-grained spatial modeling, and extensions to video-based or multimodal FAS systems.

CRediT authorship contribution statement

Seungjin Jung : Writing – original draft, Validation, Methodology, Investigation, Data curation; **Yonghyun Jeong:** Supervision, Project administration, Methodology; **Minha Kim :** Project administration, Methodology, Data curation; **Jimin Min :** Writing – original draft, Visualization, Validation, Methodology; **Youngjoon Yoo:** Writing – review & editing, Visualization, Validation, Supervision, Resources; **Jongwon Choi:** Writing – review & editing, Supervision, Project administration, Funding acquisition, Conceptualization.

Data availability

The authors do not have permission to share data.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests:

Jongwon Choi reports financial support was provided by Doosan Enerbility. Jongwon Choi reports financial support was provided by Institute of Information & Communications Technology Planning & Evaluation. If there are other authors, they declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (Ministry of Science and ICT) (RS-2021-II211341, Artificial Intelligence Graduate School Program(Chung-Ang University) and Graduate School of Metaverse Convergence support program (IITP-2023(2024)-RS-2024-00418847)).

References

[1] F. Jiang, Q. Li, B. Liu, W. Wang, C. Shan, Z. Sun, M.-H. Yang, Learning knowledge-based prompts for robust 3D mask presentation attack detection, IEEE Trans. Pattern Anal. Mach. Intell. 48 (2025) 1321–1338.
[2] A. Antil, C. Dhiman, Unmasking deception: a comprehensive survey on the evolution of face anti-spoofing methods, Neurocomputing 617 (2025) 128992.
[3] Z. Zhang, J. Yan, S. Liu, Z. Lei, D. Yi, S.Z. Li, A face antispoofing database with diverse attacks, in: International Conference on Biometrics, 2012.
[4] I. Chingovska, A. Anjos, S. Marcel, On the effectiveness of local binary patterns in face anti-spoofing, in: BIOSIG, 2012, pp. 1–7.
[5] D. Wen, A.K. Jain, H. Han, Face spoof detection with image distortion analysis, IEEE Trans. Inf. Forensic Secur. 10 (2015) 746–761.
[6] Z. Boulkenafet, J. Komulainen, L. Li, X. Feng, A. Hadid, OULU-NPU: a mobile face presentation attack database with real-world variations, in: IEEE International Conference on Automatic Face and Gesture Recognition, 2017.
[7] X. Guo, Y. Liu, A. Jain, X. Liu, Multi-domain learning for updating face anti-spoofing models, in: Proceedings of the European Conference on Computer Vision, 2022.
[8] Q. Zhou, K.-Y. Zhang, T. Yao, R. Yi, K. Sheng, S. Ding, L. Ma, Generative domain adaptation for face anti-spoofing, in: Proceedings of the European Conference on Computer Vision, 2022.
[9] Y. Sun, Y. Liu, X. Liu, Y. Li, W.-S. Chu, Rethinking domain generalization for face anti-spoofing: separability and alignment, in: Conference on Computer Vision and Pattern Recognition, 2023.
[10] K. Srivatsan, M. Naseer, K. Nandakumar, Flip: cross-domain face anti-spoofing with language guidance, in: International Conference on Computer Vision, 2023.
[11] Y. Ma, J. Qian, J. Li, J. Yang, Dual feature disentanglement for face anti-spoofing, Pattern Recognit. 155 (2024) 110656.
[12] R. Huang, X. Wang, Face anti-spoofing using feature distilling and global attention learning, Pattern Recognit. 135 (2023) 109147.
[13] Z. Wang, Q. Wang, W. Deng, G. Guo, Face anti-spoofing using transformers with relation-aware mechanism, IEEE Trans. Biom. Behav. Identity Sci. 4 (2022) 439–450.
[14] C.-H. Liao, W.-C. Chen, H.-T. Liu, Y.-R. Yeh, M.-C. Hu, C.-S. Chen, Domain invariant vision transformer learning for face anti-spoofing, in: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, 2023.
[15] Z. Wang, Z. Yu, X. Wang, Y. Qin, J. Li, C. Zhao, X. Liu, Z. Lei, Consistency regularization for deep face anti-spoofing, IEEE Trans. Inf. Forensics Secur. 18 (2023) 1127–1140.
[16] L.T. Menon, A.L. Koerich, A.S. Britto Jr, Style transfer applied to face liveness detection with user-centered models, (2019). arXiv:1907.07270
[17] O. Nikisins, A. George, S. Marcel, Domain adaptation in multi-channel autoencoder based features for robust face anti-spoofing, in: International Conference on Biometrics, 2019.
[18] Z. Wang, Z. Wang, Z. Yu, W. Deng, J. Li, T. Gao, Z. Wang, Domain generalization via shuffled style assembly for face anti-spoofing, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022.
[19] S. Yadav, A. Ross, CIT-GAN: cyclic image translation generative adversarial network with application in iris presentation attack detection, in: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, 2021.
[20] X. Long, J. Zhang, S. Shan, Generalized face liveness detection via de-fake face generator, IEEE Trans. Pattern Anal. Mach. Intell. 47 (3) (2024) 1818–1831.
[21] A. Antil, C. Dhiman, Securing faces: a GAN-powered defense against spoofing with MSRCR and CBAM, in: International Conference on Pattern Recognition, 2024, pp. 430–449.
[22] Y. Atoum, Y. Liu, A. Jourabloo, X. Liu, Face anti-spoofing using patch and depth-based CNNs, in: IEEE International Joint Conference on Biometrics, 2017.
[23] K.-Y. Zhang, T. Yao, J. Zhang, S. Liu, B. Yin, S. Ding, J. Li, Structure destruction and content combination for face anti-spoofing, in: IEEE International Joint Conference on Biometrics, 2021.
[24] W. Wang, F. Wen, H. Zheng, R. Ying, P. Liu, Conv-MLP: a convolution and mlp mixed model for multimodal face anti-spoofing, IEEE Trans. Inf. Forensics Secur. 17 (2022) 2284–2297.
[25] T. Shen, Y. Huang, Z. Tong, FacebagNet: bag-of-local-features model for multi-modal face anti-spoofing, in: Conference on Computer Vision and Pattern Recognition Workshops, 2019.
[26] C.-C. Chuang, C.-Y. Wang, S.-H. Lai, Generalized face anti-spoofing via multi-task learning and one-side meta triplet loss, in: 2023 IEEE 17th International Conference on Automatic Face and Gesture Recognition (FG), 2023, pp. 1–8.
[27] C.-Y. Wang, Y.-D. Lu, S.-T. Yang, S.-H. Lai, PatchNet: a simple face anti-spoofing framework via fine-grained patch recognition, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022.
[28] Z. Yu, R. Cai, Y. Cui, X. Liu, Y. Hu, A.C. Kot, Rethinking vision transformer and masked autoencoder in multimodal face anti-spoofing, Int. J. Comput. Vis. 132 (11) (2024) 5217–5238.
[29] Q. Yang, X. Zhu, J.-K. Fwu, Y. Ye, G. You, Y. Zhu, PipeNet: selective modal pipeline of fusion network for multi-modal face anti-spoofing, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, 2020.
[30] A. Radford, J.W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, Learning transferable visual models from natural language supervision, in: International Conference on Machine Learning, 2021.
[31] A. Liu, S. Xue, J. Gan, J. Wan, Y. Liang, J. Deng, S. Escalera, Z. Lei, CFPL-fas: class free prompt learning for generalizable face anti-spoofing, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2024, pp. 222–232.
[32] R. Cai, C. Soh, Z. Yu, H. Li, W. Yang, A.C. Kot, Towards data-centric face anti-spoofing: improving cross-domain generalization via physics-based data synthesis, Int. J. Comput. Vis. 133 (2025) 1689–1710.
[33] G. Wang, F. Lin, T. Wu, Z. Liu, Z. Ba, K. Ren, FSFM: a generalizable face security foundation model via self-supervised facial representation learning, in: Proceedings of the Computer Vision and Pattern Recognition Conference, 2025, pp. 24364–24376.
[34] T. Park, J.-Y. Zhu, O. Wang, J. Lu, E. Shechtman, A. Efros, R. Zhang, Swapping autoencoder for deep image manipulation, in: Advances in Neural Information Processing Systems, 2020.
[35] T. Karras, S. Laine, T. Aila, A style-based generator architecture for generative adversarial networks, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019.
[36] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, An image is worth 16x16 words: transformers for image recognition at scale, in: International Conference on Learning Representations, 2021.
[37] Y. Wen, K. Zhang, Z. Li, Y. Qiao, A discriminative feature learning approach for deep face recognition, in: European Conference on Computer Vision, 2016.
[38] H. Li, W. Li, H. Cao, S. Wang, F. Huang, A.C. Kot, Unsupervised domain adaptation for face anti-spoofing, IEEE Trans. Inf. Forensics Secur. 13 (2018) 1794–1809.
[39] A. Liu, Z. Tan, J. Wan, S. Escalera, G. Guo, S.Z. Li, Casia-surf cefa: a benchmark for multi-modal cross-ethnicity face anti-spoofing, in: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, 2021, pp. 1179–1187.
[40] S. Zhang, A. Liu, J. Wan, Y. Liang, G. Guo, S. Escalera, H.J. Escalante, S.Z. Li, Casia-surf: a large-scale multi-modal benchmark for face anti-spoofing, IEEE Trans. Biom. Behav. Identity Sci. 2 (2) (2020) 182–193.
[41] A. George, Z. Mostaani, D. Geissenbuhler, O. Nikisins, A. Anjos, S. Marcel, Biometric face presentation attack detection with multi-channel convolutional neural network, IEEE Trans. Inf. Forensics Secur. 15 (2019) 42–55.
[42] A. Liu, Z. Tan, J. Wan, Y. Liang, Z. Lei, G. Guo, S.Z. Li, Face anti-spoofing via adversarial cross-modality translation, IEEE Trans. Inf. Forensics Secur. (2021).
[43] X. Long, J. Zhang, S. Shan, Confidence aware learning for reliable face anti-spoofing, IEEE Trans. Inf. Forensics Secur. (2025).
[44] A. Liu, CA-moeit: generalizable face anti-spoofing via dual cross-attention and semi-fixed mixture-of-expert, Int. J. Comput. Vis. 132 (11) (2024) 5439–5452.
[45] J. Guo, A. Liu, Y. Diao, J. Zhang, H. Ma, B. Zhao, R. Hong, M. Wang, Domain generalization for face anti-spoofing via content-aware composite prompt engineering, IEEE Trans. Multimed. 28 (2025) 102–113.
[46] K. Zhang, Z. Zhang, Z. Li, Y. Qiao, Joint face detection and alignment using multitask cascaded convolutional networks, IEEE Signal Process. Lett. 23 (2016) 1499–1503.
[47] K. Zhou, J. Yang, C.C. Loy, Z. Liu, Learning to prompt for vision-language models, Int. J. Comput. Vis. 130 (9) (2022) 2337–2348.
[48] Y. Li, H. Mao, R. Girshick, K. He, Exploring plain vision transformer backbones for object detection, in: European Conference on Computer Vision, 2022, pp. 280–296.