Domain Invariant Vision Transformer Learning for Face Anti-spoofing

Chen-Hao Liao\textsuperscript{1}, Wen-Cheng Chen\textsuperscript{2}, Hsuan-Tung Liu\textsuperscript{3}, Yi-Ren Yeh\textsuperscript{4}, Min-Chun Hu\textsuperscript{5}, Chu-Song Chen\textsuperscript{1}
\textsuperscript{1}National Taiwan University, \textsuperscript{2}National Cheng Kung University, \textsuperscript{3}E.SUN Financial Holding Co., Ltd., \textsuperscript{4}National Kaohsiung Normal University, \textsuperscript{5}National Tsing Hua University
r09922113@csie.ntu.edu.tw, jerrywiston@mislab.csie.ncku.edu.tw, ahare-18342@esunbank.com.tw, yryeh@nknu.edu.tw, anitahu@cs.nthu.edu.tw, chusong@csie.ntu.edu.tw

Abstract

Existing face anti-spoofing (FAS) models have achieved high performance on specific datasets. However, for the application of real-world systems, the FAS model should generalize to the data from unknown domains rather than only achieve good results on a single baseline. As vision transformer models have demonstrated astonishing performance and strong capability in learning discriminative information, we investigate applying transformers to distinguish the face presentation attacks over unknown domains. In this work, we propose the Domain-invariant Vision Transformer (DiVT) for FAS, which adopts two losses to improve the generalizability of the vision transformer. First, a concentration loss is employed to learn a domain-invariant representation that aggregates the features of real face data. Second, a separation loss is utilized to union each type of attack from different domains. The experimental results show that our proposed method achieves state-of-the-art performance on the protocols of domain-generalized FAS tasks. Compared to previous domain generalization FAS models, our proposed method is simpler but more effective.

1. Introduction

Face recognition technology is used in many application scenarios, such as access verification in key areas, mobile phone registration and payment systems. Modern face recognition models have achieved high accuracy in face recognition. However, face presentation attacks (such as printed face photos and replayed face videos) still pose serious security risks to face recognition models, raising the need for face anti-spoofing (FAS) research.

Several approaches are proposed, including pixel-level supervision using auxiliary information and disentanglement of the spoof trace from the data, to improve the efficacy of FAS models [1, 14, 31, 38, 43]. These methods can achieve high performance on specific datasets or domains. However, even if the attack types are the same, they cannot well identify attack samples from different domains.

To make the learned model effective in different domains, various domain generalized FAS methods are introduced [4, 13, 18, 19, 27, 28, 30, 32, 41]. In the research track on conducting domain generalization models, it is assumed that the model is learned from some training domain datasets \(D_1 \cdots D_K\) and then applied to an unknown target-domain dataset \(D_{K+1}\) in a zero-sample way. That is, no target-domain data are available in the model-learning phase in either a supervised or unsupervised sense. The learned model should be insensitive to domain changes and can be successfully applied to unknown domains.

To address the above mixed-domain FAS problem, state-of-the-art domain generalization methods [13, 18, 19, 32] utilize adversarial learning, feature generation networks, meta-learning of adaptive feature normalization, or contrastive learning on Convolutional Neural Networks (CNN) backbone to extract robust features. Since the purpose of FAS is to classify whether an input face image is a real (i.e. live) face or a spoofed face, currently the leading methods [13, 32] tend to centralize all of the real faces of different domains in the feature embedding space or unifying style information related to liveness. Domain-specific or attack-type dependent representations are separated and pushed away from each other during the learning process. The feature space learned in this way can effectively generalize to unknown domains concentrated or emphasized by real face embeddings, and domain-specific attack information is distributed or suppressed.

In this paper, we propose a new approach to domain generalized FAS. Note that spoofing patterns can be globally distributed over the attacked input face image. Because transformer-based models can provide a larger receptive field than CNN and are good at capturing long-range dependencies [25], these models are better at extracting globally distributed cues, a niche for facial spoofing determination tasks. Therefore, we adopt the visual transformer architecture as the backbone in our domain generalizable FAS approach. It can take advantage of input-adaptive attention and global relational encoding that is lacking in CNNs.
In our work, we centralize the feature embedding of the real face from all domains. And all the attacked face of the same type from different domains form a separated category. However, transformer models (such as ViT [8] and swin transformer v1, v2 [21, 22]) suffer from large model size and computational resources. To address this issue, we adopt a lightweight but efficient transformer model Mobile-ViT [24] in the proposed method for domain-varying FAS.

Inspired by the works [13, 32], we also unify real faces from all domains into a group and expect to learn their feature embeddings that are invariant to this group. This enforces a uniform categorization of real or liveness face patterns regardless of their domains. However, unlike previous works that used complex adversarial training mechanisms to attain the goal, in our method, as the transformer model is already powerful in feature learning of the whole face, we only adopt a simple concentration loss to centralize the real faces in the embedding space and find that the performance on the domain generalized FAS is quite favorable.

As for the attacked faces, unlike previous work, we also unify the data of the same attack type from all domains into one category. We then use a separation loss to push groups of different attack types and real faces away from each other. Our approach is simple, easy to implement, and effective. In experiments, we collect multiple FAS datasets and apply a leave-one-out setting to evaluate the domain generalization ability of the proposed solution. The results show that our method not only outperforms existing domain-generalized FAS methods, but is also more efficient in terms of resource consumption. Figure 1 illustrates our idea, which is succinctly used to learn domain-invariant feature representations in FAS. Due to the strong capability of transformer models on learning the discriminating information that can be not only locally specific but also globally distributed, we find that simple loss and learning mechanism designs are efficient and perform reasonably well in domain-generalized FAS.

2. Related Work

The study of FAS can be characterized in terms of several aspects, including the modality of the input signal and the type of approaches (e.g. frame-based or video-based).

Multi-modality: More than one modality can be used to distinguish between real and spoofed face images. For example, we can combine 3D sensors and RGB cameras to form a multimodal FAS classifier [9]. Since not all mobile phones are equipped with powerful 3D sensors, RGB images are commonly used in recent FAS studies [40].

Frame-level vs video-level: Spoofed faces can be determined using individual image frames or from a video [20, 33, 42]. The former does not assume the availability of temporal motion information. The latter can utilize cross-frame matching or motion estimation cues to enrich feature representations and improve performance. However, video-based methods introduce more response latency time for FAS systems because they rely on grabbing a sufficient number of input frames. On the other hand, frame-level methods can be more flexibly integrated into responsive and efficient interactive systems. Yet, the problem is more challenging because only image-based information is used.

This paper introduces a new RGB-image-based method for domain generalized FAS. We give a concise review of frame-level RGB-based FAS in Sec. 2.1, and then survey vision transformer models and their usage in FAS in Sec. 2.2.

2.1. RGB Image-based FAS

Early RGB-based FAS methods exploited various handcrafted local descriptors, such as local binary patterns [5], gradient histograms [16], and speeded-up robust features [2]. The extracted features are fed into a binary classifier like a support vector machine to determine if the input image is an attack. With the success of deep learning, many methods use CNN-based models for FAS tasks. CDCN [43] and BCN [39] use depth and reflection maps generated by using other models [11, 44] to improve the discriminability of learned FAS models with pixel-wise supervision. CDCN further leverages neural architecture search (NAS) on the proposed central difference convolution to find a more powerful model and boost the performance. STDN [38] and Dual-stage Feature Learning FAS [31] employ generative adversarial training to learn models for disentangling the spoof trace from the images. The generated traces further increase the explainability of the model’s decision.

Our work focuses on domain-generalizable FAS. Although the above methods achieve good performance when the training and testing domains have little distribution shift, they show poor generalization ability if there is a large discrepancy among the domains. As a consequence, many domain generalized FAS methods have been proposed. SSDG [13] uses single-side adversarial training to make the extracted features of real data more invariant across different
domains. Also, an asymmetric triplet loss is proposed to aggregate the features of the same classes (real data of all domains and spoof data of separated domains) and scatter these classes. ANRL [19] explores refining the normalization mechanism in the feature extraction process to improve the domain-generalization ability. Adaptive normalization is proposed to enforce the model to extract domain-agnostic and discriminating representation for the face images. SSAN [32] introduces the use of content and style disentanglement to solve the FAS problem. The approach extracts the style features of the face images and then applies contrastive learning to extract the generalized representation across different domains. FGHV [18] proposes to generate different distribution hypotheses of real faces and known attacks. By fitting the face feature to the hypothesis generated by the feature generation network with the Gaussian input, the extracted features are more reliable in defending against attacks in unknown domains.

2.2. Transformers and FAS

Transformer [29] has been widely used in natural language processing and has gained more attention in solving computer vision tasks. Dosovitskiy et al. [8] proposed the Vision Transformer (ViT), instead of treating pixels as tokens in a self-attention mechanism, it divides the image into many patches and projects them into a low-dimensional feature space to make the computation affordable. Later, a lot of work improved the ViT model. Swin Transformer [22] introduces a shifted-window attention mechanism, which computes self-attention within a local window and simulates cross-region relations by shifting windows in successive layers. Focal Transformer [37] proposes focal self-attention. Each patch focuses not only on other patches in the local window, but also on the summarized tokens outside to encode long-range information with marginal overhead. CoAtNet [6] considers the similarity in computational form between self-attention and depth-wise convolution. They fuse the two modules by adding input-independent weights to the attention mechanism, embedding translation-equivalent information into the transformer. MobileViT [24] combines convolution and transformer in one module to capture local and global information efficiently. With the utilization of this module, the model provides good performance even if the model is shallow and makes the visual translator more suitable for edge devices.

In the past, only a few studies have used the transformer model in FAS [10, 12]. The approach in [10] directly uses ViT [8] with binary cross-entropy loss for FAS. Unlike [10], the method in [12] uses the transformer models in an indirect way; it adopts multiple visual transformers as the teacher model, and aims to train a smaller student CNN and improve the student model’s performance. Thus the solution is still a CNN inference model. Apart from the issue of computational overhead, although they can achieve competitive performance in the single-domain setting, they are not designed to handle domain generalized FAS problems.

Instead, our work uses a light-weight transformer model, MobileViT [24], which contains fewer parameters. Leveraging the transformer models, we propose two loss terms to handle the cross-domain FAS problem, domain-invariant concentration loss and attack separation loss. Our solution, referred to as Domain-invariant Vision Transformer (DiVT) for FAS, can achieve higher performance than the previous approaches on the domain generalized FAS problem with comparable or better resource consumption efficiency.

3. Proposed Method

Our approach takes a transformer model as the network backbone. Without loss of generality, we employ MobileViT [24] as the backbone model of our approach. It can be replaced with the other transformer models as well (eg., ViT [8], Swin Transformer [22]). In the experiments, we present our study on the ablation results of choosing the backbone transformer model for domain generalized FAS.

Our employed MobileViT is composed of a series of MobileNet-v2 [26] and MobileViT blocks. The MobileNet-v2 blocks are primarily responsible for down-sampling the feature maps. The MobileViT block models the spatial relationships, where the feature map is first processed by a convolution layer (to encode the local spatial information) and a point-wise convolution (to project into a high-dimensional space). It is then partitioned into a sequence of patches fed into multiple transformer modules to encode the global relationships. Later, further projection and fusion are applied before producing the output. Details can be found in [24].

3.1. Domain-invariant Concentration Loss

Suppose we have $K$ datasets, namely, $\mathcal{D}_1, \cdots, \mathcal{D}_K$; each dataset specifies a domain. Assume that one domain contains $C$ types of attacks, and $\mathcal{D}_k^c$ denotes the dataset consisting of the $c$-th type attack images in domain $k$ where $k \in \{1 \cdots K\}$ and $c \in \{1 \cdots C\}$. In addition, let $\mathcal{D}_k^{\text{real}}$ indicate the set of real face images in domain $k$.

Given an actual face image in $\mathcal{D}_k^{\text{real}}$, our goal is to provide it with a feature representation that is not biased toward specific domains. The learned representation is thus expected to be invariant to the domain changes. To achieve this purpose, we simply union all the real faces of different domains as a positive (non-spoof) class of data as follows:

$$\mathcal{D}_k^{\text{R}} = \bigcup_{k=1}^{K} \mathcal{D}_k^{\text{real}}.$$  \hfill (1)

When passing the data in $\mathcal{D}_k^{\text{R}}$ to a deep transformer model $\pi$ (e.g., MobileViT), let $\mathbf{E}_k^{R} = \pi(\mathcal{D}_k^{R})$ be the feature representations obtained in the embedding layer. That is, we join
all domains’ real face embedding as a group $E^R$. Then, we hope that $E^R$ is concentrated on the origin of the feature embedding space, $0 = [0]^d$ (the $d$-dimensional vector with all elements being zero), where $d$ is the dimension of the feature embedding space of the transformer model $\pi$.

Hence, no matter the domain of a real face image, we hope that its feature embedding is near to the origin of the embedding space. The idea of pulling the features to the origin has also been used for action analysis [17]. The domain-invariant concentration (DiC) loss is defined as follows.

$$L_{DiC} = \frac{1}{N} \sum_{i=1}^{N} \mathbb{1}[x_i \in D^R] \cdot \|f_i\|_1,$$  \hspace{1cm} (2)

where $\mathbb{1}$ is the indicator function, ‘$\cdot$’ denotes the inner product, $N$ means the batch size, and $f_i = \pi(x_i)$ is the $i$-th feature embedding extracted by the transformer backbone, respectively. In essence, Equation 2 encourages to make smaller the norm of the feature embedding learned for the real face images in all domains. An illustration is shown in the right bottom part of Figure 2.

It is worth comparing our concentration loss $L_{DiC}$ with the center loss [35] widely used for effective training of a face recognizer (FR). In the center loss, each category has a center. When giving a sample, we hope to make the feature embedding close to the center of the category that contains this sample. As each individual defines a category in FR, multiple categories exist and their centers have to be learned together with the network weights. However, in our domain-generalized FAS, the real-face patterns are unified while the ways of attack types have infinite possibilities. We thus merely center the features of real face and let the spoofing features distributed in the space freely. Since we only apply the centering principle to a single category (real face), it is unnecessary to express multiple group centers simultaneously. Hence, we can skip the parametrization for learning of the group centers and directly specify the center at the origin. The center does not move with mini-batches and the training process is easier and stable.

### 3.2. Domain-invariant Attack-separation Loss

The concentration loss encourages the real-face embedding to have smaller norms and pulls all their features to the origin. For each type of attack, we also hope to group the data belonging to the attack regardless of the data’s domain. To this end, we also group all domains’ spoofed faces of the same attack type as follows:

$$D^c = \bigcup_{k=1}^{K} D^c_k, \ c \in \{1 \cdots C\}.$$  \hspace{1cm} (3)

As the origin can draw the actual face features in the embedding space, no matter the domains, we hope to push the attack images’ feature representation to each other and away from the origin. Figure 1 illustrates the idea. To achieve this purpose, We simply add a classification layer in the transformer model $\pi$ to classify the data into the categories of real face and different attack types via cross-entropy loss.

Consider a batch consisting of $N$ samples $\{x_1 \cdots x_N\}$. Let $\hat{y}_i = \mathbb{1}[x_i \in D^c] C_{c=0}$ be the corresponding domain-union one-hot label of $x_i$, where $D^0 (c = 0)$ represents
the real face category $D^R$ to simplify the notation. The
domain-invariant attack-separation loss is defined as:

$$L_{DiA}^{ce} = \frac{1}{N} \sum_{i=1}^{N} \sum_{c=0}^{C} -y_{ic}^c \log y_{i}^c,$$  \hspace{1cm} (4)

where $y_{ic}^c$ is the class $c$'s softmax output produced by
the transformer model $\pi$. The attack types classification task
separate the groups of different attack types and real faces
from each other. It enforces the model to learn a domain
insensitive latent space.

3.3. Training and Testing

In the training phase, we train the transformer model by
combining the two losses in a supervised manner. A hyper-
parameter $\lambda$ is used as a balance factor between them.

$$L_{total} = L_{DiA}^{ce} + \lambda L_{DiC}$$ \hspace{1cm} (5)

By shrinking the real-face feature embedding toward the
origin and separating different types of the attacked embed-
ding in a transformer model, our approach is simple but ef-
effective in learning domain-invariant representations to solve
the associated FAS problem.

Figure 2 gives an overview of our approach, DiVT for
FAS. In the testing phase, we directly use the output of Real
head (in Figure 2) as the predicted probability of the input
image captured from a real person. Our approach is easy
to realize and can achieve state-of-the-art performance on
standard benchmarks in domain-generalized FAS. Experimental results demonstrate the efficacy of our method.

4. Experiments

4.1. Datasets and Evaluation Metrics

We evaluate our method using four public FAS datasets,
namely, CASIA-FASD [45], MSU-MFSD [34], Idiap
is collected by using three cameras with different video
qualities under natural scenes. Print and replay attacks
are produced by printing the highest-quality image on cop-
per papers and displaying the videos on a tablet, respect-
ively. MSU-MFSD is collected by using a laptop and a
mobile phone camera. Two qualities of replay attacks are
introduced by playing a high-end camera-recorded video
on a tablet and a mobile-recorded video on another mobile
phone. The high-quality photos are printed on paper to pro-
duce print attacks. Idiap Replay-Attack is gathered under
two different environments, a lamp illuminated one with a
uniform background and a day-light illuminated one with a
complex scene. The replay and print attacks are generated
by a similar setting to the MSU-MFSD dataset with differ-
ent devices. Besides, these attack materials are held either
by hands or via a fixed-support. OULU-NPU is collected
during three sessions with different illuminations and back-
grounds. The videos are recorded using six different mobile
phones. Two printers and two video players are utilized to
simulate the diversity of devices the intruder will use.

Following the setting of domain-generalized FAS [13],
we only use the training and testing sets in Idiap Replay-
Attack and OULU-NPU, while discarding their validation
sets. The other two datasets are all used. Table 1 shows
the amount of real and fake videos utilized in our experi-
ment. The Half Total Error Rate (HTER) and the Area Un-
der Curve (AUC) are utilized as the evaluation metrics.

### Table 1. The number of real and fake videos used in our evaluation.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Real videos</th>
<th>Fake videos</th>
</tr>
</thead>
<tbody>
<tr>
<td>CASIA-FASD</td>
<td>150</td>
<td>450</td>
</tr>
<tr>
<td>MSU-MFSD</td>
<td>70</td>
<td>210</td>
</tr>
<tr>
<td>Idiap Replay-Attack</td>
<td>140</td>
<td>700</td>
</tr>
<tr>
<td>OULU-NPU</td>
<td>720</td>
<td>2880</td>
</tr>
</tbody>
</table>

4.2. Implementation Details

In the image pre-processing stage, we align all the video
frames by MTCNN [36] algorithm. We then crop the face
regions, and resize the cropped regions into $256 \times 256$.

Because there is little discrepancy among different
frames in a video, we follow the same training setting as
[13], which randomly samples one frame in each video as
the training data. In each training step, the same number
of real and fake data are sampled from all training datasets.

We use MobileViT-S [24] implemented by CVNets [23]
as our backbone. The model is pre-trained on ImageNet-1K
[7] and optimized by Adam optimizer [15] with the learn-
ing rate and weight decay parameter being $10^{-4}$ and $10^{-6}$,
respectively. The balance factor $\lambda$ is set to 0.2 in our work.

4.3. Domain Generalized Evaluation

4.3.1 Leave-one-out setting

To evaluate the approaches in domain generalized FAS, a
commonly adopted setting is the leave-one-out testing on
the datasets mentioned in Section 4.1. In this evaluation
protocol, the model is trained on three of the datasets and
then tested on the remaining dataset. We follow the setting
and show the performance comparison of our approach and
previous competitive methods in Table 2 (each dataset is de-
noted using its prefix). Note that the methods are all frame-
level approaches like ours, except that NAS-FAS [42] is a
video-based approach that utilizes further temporal motion
information to enhance performance.

The results shown in Table 2 for comparison refer to the
papers of SSAN [32] and NAS-FAS [42]. The best-
and second-best- performed methods are shown in bold
and underline, respectively. Among the previous methods,
SSAN-R is the state-of-the-art model and NAS-FAS has outstanding performance on some evaluation sets. Compared to the previous state-of-the-art domain generation methods of FAS, such as SSDG and SSAN, our proposed DiVT achieves better performance on all evaluation sets. The improvement of HTER in our work is particularly significant. There are two settings that even improve more than 3%. The results show that our approach is a more favorable one than the previous approaches.

The only evaluation result where our method achieved second place is the AUC measure setting I&C&M to O. The best model on this evaluation set is NAS-FAS, but its performance on HTER is not as good as ours. However, NAS-FAS is a video-based method. In contrast, our DiVT-M, an image-based method, still achieves competitive results (with a difference of less than 0.2% in AUC).

### 4.3.2 Limited training data setting

The above protocol uses larger-scale training domain data for the performance comparison. Another popular setup is to use smaller-scale training domain data for the evaluation.

We also evaluate our method while the training data is limited in the setting (following [13]). The MSU-MFSD and Replay-Attack datasets are used as training data, and the two remaining datasets are used as testing data. Since SSDG-R [13] and SSAN-R [32], which use a stronger convolutional backbone, are more effective models than the other respective versions in the works [13] and [32]. For a fairer comparison, we use the source codes released for SSDG-R and SSAN-R to re-train this setting and obtain better results than that achieved using weaker backbone models shown in [13] and [32], respectively. As can be seen in Table 3, our method still demonstrates its effectiveness in the situation of limited training data and outperforms previous state-of-the-art domain generalization methods in general. The only result our method performs worse is the HTER in the M&I to C setting (0.25% worse than SSDG-R). However, our method is still better in AUC (0.25% higher). Since AUC generally reflects the balance between false acceptance and rejection with varying thresholds, a higher AUC reveals that our method is generally better.

### 4.4. Ablation Study

We conduct several ablation studies to evaluate our proposed method, including using different backbones, the efficacy of proposed losses, different classification objectives, and combining domain adversarial training.

#### 4.4.1 Different Backbones

We evaluate the performance of our approach using different vision transformer backbones, including vanilla ViT (ViT-Base) [8], Swin Transformer (Swin-T) [22], and MobileViT (MobileViT-S) [24]. They are denoted as DiVT-V, DiVT-S, and DiVT-M, respectively. We also evaluate our method on ResNet-18, a CNN backbone to compare the effectiveness of using CNN and transformer. All of the backbones are pre-trained on ImageNet-1K dataset. We adopt hyper-parameter tuning to find the best balance factor $\lambda$ for four backbones. The factor we use are 0.5, 0.05, 0.2, and 0.2, respectively. Table 4 shows the results, and the upper half shows the results when these backbones are trained by using binary cross-entropy loss only.

The results reveal that transformer backbones mostly perform better than CNN. The superiority in performance could be due to the attention module and global feature-
Table 4. Performance on the domain-generalized evaluation of the proposed method with various backbones. The suffixes after DiVT represent the adopted feature extractor: ResNet-18, ViT, ViT(Tiny), Swin Transformer, and MobileViT, respectively. The upper half shows the results when these backbones are trained by using binary cross-entropy only.

<table>
<thead>
<tr>
<th>Methods</th>
<th>O&amp;C&amp;I to M</th>
<th>O&amp;M&amp;I to C</th>
<th>O&amp;C&amp;M to I</th>
<th>I&amp;C&amp;M to O</th>
<th>Average</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>HTER(%)</td>
<td>AUC(%)</td>
<td>HTER(%)</td>
<td>AUC(%)</td>
<td>HTER(%)</td>
</tr>
<tr>
<td>ResNet-18</td>
<td>12.62</td>
<td>93.78</td>
<td>25.89</td>
<td>84.67</td>
<td>25.00</td>
</tr>
<tr>
<td>ViT-Base</td>
<td>7.14</td>
<td>97.94</td>
<td>24.00</td>
<td>84.27</td>
<td>10.79</td>
</tr>
<tr>
<td>ViT-Tiny</td>
<td>8.57</td>
<td>97.18</td>
<td>22.00</td>
<td>86.85</td>
<td>15.00</td>
</tr>
<tr>
<td>Swin-T</td>
<td>2.86</td>
<td>99.34</td>
<td>11.78</td>
<td>95.83</td>
<td>11.36</td>
</tr>
<tr>
<td>MobileViT-S</td>
<td>5.48</td>
<td>93.99</td>
<td>13.22</td>
<td>93.32</td>
<td>17.14</td>
</tr>
<tr>
<td>DiVT-ResNet</td>
<td>11.43</td>
<td>94.68</td>
<td>18.67</td>
<td>91.32</td>
<td>21.43</td>
</tr>
<tr>
<td>DiVT-V</td>
<td>10.00</td>
<td>96.64</td>
<td>14.67</td>
<td>93.08</td>
<td>5.71</td>
</tr>
<tr>
<td>DiVT-V(Tiny)</td>
<td>7.14</td>
<td>98.27</td>
<td>11.89</td>
<td>95.17</td>
<td>11.43</td>
</tr>
<tr>
<td>DiVT-S</td>
<td>8.57</td>
<td>97.29</td>
<td>7.22</td>
<td>98.13</td>
<td>6.43</td>
</tr>
<tr>
<td>DiVT-M</td>
<td>2.86</td>
<td>99.14</td>
<td>8.67</td>
<td>96.92</td>
<td>3.71</td>
</tr>
</tbody>
</table>

Table 5. Evaluation of each component in our method. The binary classification is used while $L_{DiA}$ is not applied.

<table>
<thead>
<tr>
<th>Components</th>
<th>O&amp;C&amp;I to M</th>
<th>O&amp;M&amp;I to C</th>
<th>O&amp;C&amp;M to I</th>
<th>I&amp;C&amp;M to O</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>HTER (%)</td>
<td>AUC (%)</td>
<td>HTER (%)</td>
<td>AUC (%)</td>
</tr>
<tr>
<td>$L_{DiA}$</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>✓ ✓</td>
<td>5.48</td>
<td>93.99</td>
<td>13.22</td>
<td>93.32</td>
</tr>
<tr>
<td>✓ ✓</td>
<td>2.62</td>
<td>99.10</td>
<td>9.33</td>
<td>96.40</td>
</tr>
<tr>
<td>✓ ✓</td>
<td>5.71</td>
<td>98.36</td>
<td>10.00</td>
<td>96.80</td>
</tr>
<tr>
<td>✓ ✓</td>
<td>2.86</td>
<td>99.14</td>
<td>8.67</td>
<td>96.92</td>
</tr>
</tbody>
</table>

4.4.2 Loss Combinations and Classification Objectives

We investigate the effectiveness of two core components ($L_{DiA}^{ce}$ and $L_{DiC}$) in our method, and the results of four component combinations are illustrated in Table 5. When both of the $L_{DiA}^{ce}$ and $L_{DiC}$ are not used, we employ the classification head of two classes (real and spoof) instead, and train the model by using binary cross-entropy loss.

The results prove that both components are effective for improving the vision transformer on domain-generalized FAS tasks. The domain-invariant attack-separation loss provides the main improvement (roughly 3.7% AUC on average) and the domain-invariant concentration loss boost extraction characteristic of the transformer. Furthermore, we find that the methods using our losses (the lower half of Table 4) are generally better than the methods using the binary cross-entropy loss (the upper half of the table) in most cases. This reveals the effectiveness of our losses in overall.

As for the comparison of using different vision transformer backbones in our approach (the lower half of Table 4), we find that DiVT-V performs worse than the others. We conjecture the reason to be that ViT lacks the modeling of local patterns and has a huge number of parameters, which requires a large amount of training data to converge. Swin Transformer and MobileViT adopt hierarchical architecture or convolutional modules to model the local spatial property, which can adapt to the situation of less training data. Both of these two methods achieve competitive performance. Since DiVT-M achieves the best average performance on both evaluation metrics and has the smallest model size, we use it in the following studies.

Size-compatible ViT comparison: DiVT-M performs better than DiVT-V. This could be due to the appropriate ratio of the model size to the amount of training data. Hence, we further investigate the performance of using ViT-Tiny [8] as the backbone, which has a comparable model size with DiVT-M. As shown in Table 4, DiVT-V(Tiny) outperforms DiVT-V probably because of its suitable size for the data. DiVT-M still achieves the best among the transformer models. We conjecture that it is because MobileViT also takes the advantage of convolution, which is lacking in the others.

Comparison to FAS using transformer [10]: Only a few works [10, 12] have applied transformers for FAS. Since [12] mainly uses transformers as teacher models for distillation and still conducts a CNN model for inference, we compare [10] in the experiments. As mentioned before, [10] just adopts ViT as the backbone with binary cross-entropy loss. Hence, the results of ViT-Base in Table 4 just reveal its performance on the leave-one-out domain-generalized FAS protocol. As can be seen, ViT-Base [10] performs worse than DiVT-V in most cases. When replacing the backbone with ViT-Tiny, Swin-T, and MobileViT-S, their average performances are still worse than DiVT-V(Tiny), DiVT-S, and DiVT-M, respectively. Another version of implementation in [10] is to fix the backbone weights and train the classifier layer only. We have done the experiments too, but the results are far worse and are shown in the supplementary material. From the results, our method is more favorable.
Table 6. Performance of different categorized methods (with $L_{DiC}$ is adopted).

<table>
<thead>
<tr>
<th>Classification Objective</th>
<th>O&amp;C&amp;I to M</th>
<th>O&amp;M&amp;I to C</th>
<th>O&amp;C&amp;M to I</th>
<th>I&amp;C&amp;M to O</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>HTER(%)</td>
<td>AUC(%)</td>
<td>HTER(%)</td>
<td>AUC(%)</td>
</tr>
<tr>
<td>Binary Classification</td>
<td>5.71</td>
<td>98.36</td>
<td>10.00</td>
<td>96.80</td>
</tr>
<tr>
<td>Attack Types</td>
<td>2.86</td>
<td>99.14</td>
<td>8.67</td>
<td>96.92</td>
</tr>
<tr>
<td>Domains</td>
<td>5.95</td>
<td>98.31</td>
<td>9.89</td>
<td>96.54</td>
</tr>
<tr>
<td>Attack Types + Domains</td>
<td>9.76</td>
<td>96.37</td>
<td>12.78</td>
<td>96.12</td>
</tr>
</tbody>
</table>

Table 7. Leveraging domain-adversarial learning technique in our approach.

<table>
<thead>
<tr>
<th>Method</th>
<th>O&amp;C&amp;I to M</th>
<th>O&amp;M&amp;I to C</th>
<th>O&amp;C&amp;M to I</th>
<th>I&amp;C&amp;M to O</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>HTER(%)</td>
<td>AUC(%)</td>
<td>HTER(%)</td>
<td>AUC(%)</td>
</tr>
<tr>
<td>DiVT-M</td>
<td>2.86</td>
<td>99.14</td>
<td>8.67</td>
<td>96.92</td>
</tr>
<tr>
<td>DiVT-M + Domain-adversarial</td>
<td>4.29</td>
<td>98.20</td>
<td>7.33</td>
<td>97.56</td>
</tr>
</tbody>
</table>

Table 8. Comparison of computation resource.

about 1.3% average AUC. The model achieves the best performance while both components are applied.

In this work, the attack-separation loss is proved to be effective for the task of cross-domain FAS tasks. Based on the success, we are curious about the effect of different classification objectives on model improvement. In addition to binary classification and our attack types classification, we also conduct experiments on domain classification. Table 6 shows the results of different classification objectives, where “Domains” means categorizing the data to the real face and different domains of attacks, and “Domains + Attack Types” indicates categorizing the data into real face and the combination classes of domains and attack type. We can observe that attack types classification gains the best average performance, revealing the efficacy of the domain-invariant assumption in our approach. Domain classification improves the model a little but not significantly. The performance gets worse when adopting the combination of attack type and the domain classification. The reason may be that the model overfits on these combined categories.

4.4.3 Domain-adversarial Learning

We additionally employ the same domain adversarial loss used in both SSDG and SSAN [13, 32] to our feature extractor, which discriminates the attack domains using a gradient reversal layer and a two-layer discriminator. The results are shown in Table 7. Adding adversarial loss performs slightly worse. MobileViT still performs the best even when using the simpler losses designed in our solution. It could be because the features can already be well extracted by supervised learning. Adversarial training seems to result in an over-competition in this case. Furthermore, how to well employ vision transformers in adversarial training is still worth exploring.

4.5. Comparison of computation resources

We compare the model size (number of parameters) and FLOPs between previous methods and ours. As shown in Table 8, DiVT-M performs more favorably and requires fewer parameters than DiVT-S and DiVT-V. The model DiVT-V(Tiny) has fewer FLOPs, but its performance is worse and requires more parameters. This verifies again that the MobileViT model adopted in our approach is suitable for the domain-generalized FAS task.

5. Conclusion

Handling the attack sample from unknown domains is an important problem in face anti-spoofing. We propose Domain-invariant Vision Transformer (DiVT) to solve the domain generalized FAS problem in this work. We apply an efficient vision transformer-based module to extract both the globally and locally distributed cues of spoofing patterns. Then we introduce two loss terms to learn a domain-invariant latent space. First, a domain-invariant concentration loss is applied to concentrate the features of real faces. Second, a separation loss is adopted to push away the groups of different attack types and real faces from each other. The experimental results show that our proposed model achieves state-of-the-art performance on the cross-domain evaluation protocols. Compared to previous domain generalized FAS methods, our proposed DiVT for FAS is not only efficient and easy to implement. It is also more favorably performed.

Acknowledgement. This work was supported in part by E.SUN Financial Holding and MOST 110-2634-F-002-050.
References


[21] Ze Liu, Han Hu, Yonglin Lin, Zhuliang Yao, Zhenda Xie, Yixuan Wei, Jia Ning, Yue Cao, Zheng Zhang, Li Dong, Furu Wei, and Baining Guo. Swin transformer v2: Scaling up capacity and resolution. In International Conference on Computer Vision and Pattern Recognition (CVPR), 2022.


[27] Rui Shao, Xiangyuan Lan, Jiawei Li, and Pong C. Yuen. Multi-adversarial discriminative deep domain generalization for face presentation attack detection. In The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2019.


