The best paper work.

Face Presentation Attack Detection Using Taskonomy Feature

Anonymous CVPR submission

Paper ID 3924

Abstract

*The robustness and generalization ability of Presentation Attack Detection (PAD) methods is critical to ensure the security of Face Recognition Systems (FRSs). However, in the real scenario, Presentation Attacks (PAs) are various and hard to be collected. Existing PAD methods are highly dependent on the limited training set and cannot generalize well to unknown PAs. Unlike PAD task, other face-related tasks trained by huge amount of real faces (*e.g.* face recognition and attribute editing) can be effectively adopted into different application scenarios. Inspired by this, we propose to apply taskonomy (task taxonomy) from other face-related tasks to solve face PAD, so as to improve the generalization ability in detecting PAs. The proposed method, first introduces task specific features from other face-related tasks, then, we design a Cross-Modal Adapter using a Graph Attention Network (GAT) to re-map such features to adapt to PAD task. Finally, face PAD is achieved by using the hierarchical features from a CNN-based PA detector and the re-mapped features. The experimental results show that the proposed method can achieve significant improvements in the complicated and hybrid datasets, when compared with the state-of-the-art methods. In particular, when trained using OULU-NPU, CASIA-FASD, and Idiap Replay-Attack, we obtain HTER (Half Total Error Rate) of 5.48% in MSU-MFSD, outperforming the baseline by 7.39%. Code will be made publicly available.*

1. Introduction

Face Recognition Systems (FRSs) are widely deployed in authentication applications especially in access control and mobile phone unlocking in our daily life. However, recent studies [1, 2] demonstrate that the existing face recognition systems are lack of robustness, since they are easily spoofed by presentation attacks (PAs), such as photographs, video replays, and low-cost artificial masks [3]. Meanwhile, the face images can be easily obtained from social media, which seriously increases the risk of PAs. These issues raise wide concerns about the vulnerability of facial recognition technologies. Consequently, it is crucial to detect presentation attack to achieve robust and reliable FRS.

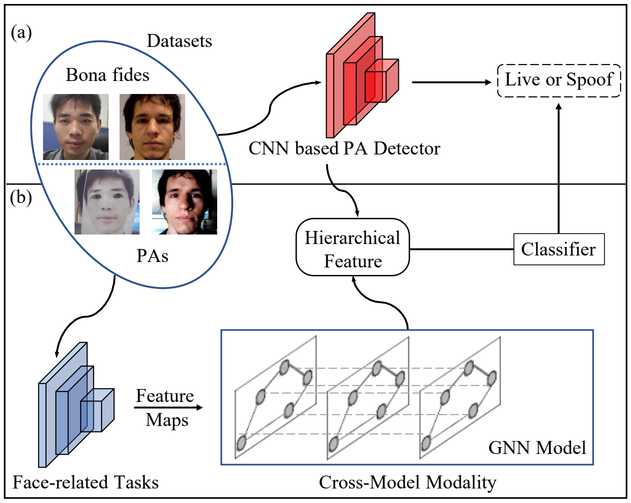


Figure 1. The diagram of (a) existing face PAD model and (b) the proposed PAD scheme. In the proposed method, face-related tasks, including face, expression and attribute recognition, are adopted as an auxiliary branch to provide robust task specific features. Then a cross-model modality, *i.e*. GNN model, are adopted to obtain taskonomy representation from task specific features to facilitate PAD.

To tackle such challenge, many face PAD methods have been proposed, which can be divided into hardware and software based methods. Hardware based solutions [4, 5] generally employ specific sensors to acquire presentations with different image modalities to detect PAs. Although, these methods are in strong security, their applicability is still limited because of unsatisfying performance to new application scenarios and cost limitations. Software based algorithms usually explore the distinctive features between bona fides (live faces) and PAs, such as hand-crafted features [6–9] and deep features [10, 11]. Due to the advances of deep learning in recent years, deep feature based methods have been widely used in the community, since better performance can be achieved by adopting convolutional neural networks (CNNs) [12–14].

However, existing CNN-based methods often need comparable bona fide and PA samples for training. In the real scenario, PAs with different materials and instruments are hard to collect. Existing PAD methods might not obtain ideal generalization to different attacks due to the limited training data. On the contrary, the amount of live faces are huge, and the samples can be conveniently collected. In many cases, the public datasets with specific face-related task (*i.e*. face recognition datasets) contain millions of faces cross genders, ages and races, which brings the strong capacity to the model.

Motivated by this consideration, we argue that PAD task should share some common patterns with the other face-related tasks, and the performance of PA detection might be boosted by the features from such tasks. Thus, in this paper, we propose a taskonomy-driven face PAD method, denoted as **TOD**. Different from existing PAD methods (shown in Fig. 1(a)), the proposed solution attempts to improve the generalization of PAD model by introducing other face-related tasks. As shown in Fig. 1(b), we design an auxiliary branch to extract task specific features from other face-related tasks. By following step-by-step graph, a Graph Neural Network (GNN) is then put forward to re-map and adapt to PAD task. The generalization capability is finally improved by alleviating the problem resulting in limited PA samples for training. The main contributions of this work are concluded as follows,

- Existing PAD methods trained on the limited datasets are vulnerable to unseen PAs. To address this problem, face-related task is introduced into face PAD to acquire task specific features, which can improve generalization ability of PAD model.
- A Cross-Modal Adapter is designed to obtain taskonomy features, which can adapt task specific features into PAD space.
- The effectiveness and superiority of our method are validated on the public datasets. Particularly, when OULU-NPU [3], CASIA-FASD [15], and Idiap Replay-Attack [16] are adopted as training set and MSU-MFSD [7] is used as test set, the HTER (Half Total Error Rate) of the proposed method can outperform the baseline by 7.39%.

2. Related Works

As face-related work is for the first time introduced in face PAD and cross-modal learning is adopted to achieve taskonomy, the proposed face PAD scheme is different from existing PAD methods. Hence, our reviews mainly include face PAD methods, face-related works, and multi-modal learning.

2.1. Face Presentation Attack Detection

Existing face PAD methods can be categorized into hand-crafted and deep learning based methods. Hand-crafted methods employ the algorithms, *e.g*. LBP [6], IDA [7], SIFT [8], and SURF [9] to extract the features and then adopt traditional classifiers such as LDA and SVM to detect PAs. However, the hand-crafted features can be easily influenced by the variations of imaging quality and illumination. As a result, feature based methods generally can not generalize well to different application scenarios.

To address such challenges, deep learning models are then proposed for face PAD. Yang et al. [10] proposed to use CNNs to extract deep discriminative features for face PAD. Nguyen et al. [17] designed a multi-task learning model, which locates the most important regions of the input to detect PAs. Yu et al. [18] proposed a Central Difference Convolution (CDC) structure to capture intrinsic detailed patterns for face PAD and then used Neural Architecture Search (NAS) in CDC based network to achieve a better result. Besides applying only RGB images, auxiliary information of face, *e.g*. face depth, are considered to establish a more robust detector. Liu et al. [19] explored face depth as auxiliary information and estimated rPPG signal of RGB images through a CNN-RNN model for face PAD. George et al. [14] introduced a cross-modal loss function to supervise the multi-stream model, which extracted features from both RGB and depth channels. Although such deep learning methods can achieve better PAD performance, their dependence on training data would inevitably leads a bias when accessible data is limited. In particular, numerous studies [20–22] have shown that, PA detector trained in one dataset can not generalize to other datasets effectively.

To improve the generalization, researchers have further proposed one-class and domain generalization methods. A one-class multi-channel CNN model [20] was proposed to learn the discriminative representation for bona fides within a compact embedding space. Different from one-class methods, domain generalization based methods pay more attention to the disparities among the domains. Shao et al. [21] proposed a multi-adversarial deep domain generalization framework to learn a generalized feature space within the dual-force triplet-mining constraint. Since the learned feature space is discriminative and shared by multiple source domains, the generalization to new face PAs can be ensured effectively. Wang et al. [22] disentangled PAD informative features from subject-driven features and then designed a multi-domain learning based network to learn domain-independent features cross different domains for face PAD. Generally speaking, when training data is adequate for the aforementioned method, the detector can achieve very competitive performance. However, unlike other face-related tasks, Presentation Attacks (PAs) are hard to collect and the types/instruments of attacks are consis-

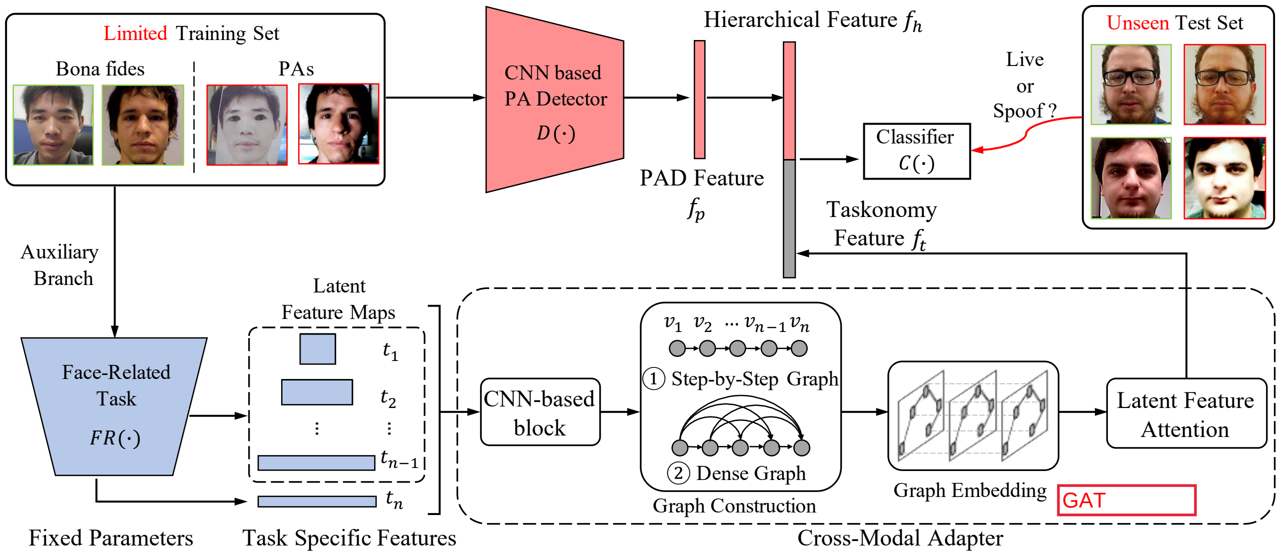


Figure 2. The pipeline of our proposed taskonomy-driven PAD method (**TOD**). In auxiliary branch, the task specific features t_i can be extracted by the parameter-fixed model $FR(\cdot)$, which has been trained from face-related tasks. Then, a CNN-based block is used to transform t_i to graph vertexes. We construct a Graph Attention Netwok (GAT) to re-map t_i to fit PAD. By following latent feature attention, the taskonomy feature f_t is obtained, and finally fused with the main branch to achieve face PAD.

tently increasing. Due to such open challenges, we propose a taskonomy-driven PAD method to decrease the dependence of the PA detector on data scale. In particular, we extract taskonomy features from other face-related tasks for a more robust representation with better generalization capability. Benefit from extensive samples collected in other tasks, face PA detector can achieve significant improvement within limited PAD data.

2.2. Face-related Tasks

With the advances in deep learning, face related tasks including face recognition, face expression recognition, face attribute editing, etc, have become a very active field [23–25]. In this section, we briefly introduce some representative tasks adopted in this work due to the limited scope of the paper. For large-scale face recognition, Deng et al. [23] proposed an Additive Angular Margin Loss (ArcFace) , which has a clear geometric interpretation, to obtain highly discriminative features. ArcFace is a solid work evaluated on the various face recognition benchmarks, including image datasets with trillions of pairs and a large-scale video dataset. In the terms of facial expression recognition, Wang et al. [24] proposed a Self-Cure Network (SCN) to address uncertainties in facial expressions. By combining self-attention and relabelling mechanism, such method can prevent deep networks from over-fitting uncertain facial images. Karras et al. [25] proposed an alternative generator architecture, called StyleGAN, to achieve automatic learning, and unsupervised separation of high-level attributes (*e.g.* pose and identity when trained on human faces). StyleGAN can intuitively control the generation with some scale-specific information, which is always set as a famous baseline in generative task. As the representative works for the corresponding tasks, the aforementioned solutions are conducted in this paper to investigate the relationship between PAD and the other face-related tasks.

2.3. Multi-modal Learning and Taskonomy

Multi-modal learning are generally applied in various tasks. For visual question answering (VAQ) tasks, two modalities, image and question in natural language, are used to infer the correct answer [26]. Video-based tasks generally requires model to separate speech in audio with specific visual target [27, 28]. However, existing multi-modal learning methods are only adopted to learn different modal representation of the same task. The relationship between modals from different tasks has not been explored. Zamir et al. [29] proposed to adopt task taxonomy (taskonomy) to predict the relationship between source visual tasks and a target visual task. They presented that source visual tasks can be well generalized to their related tasks. Dwivedi et al. [30] proposed to use efficient taskonomy and transfer learning to assess the relationship between visual tasks and their task-specific models. They proved that through learning similarities from task-specific models, the source model and training dataset size will not play a significant role. Inspired by taskonomy, in this paper, we design a Cross-Modal Adapter using Graph Neural Network (GNN) to transform features from other face-related tasks to face PAD task. In this way, we can obtain a taskonomy feature with great generalization capability for face PAD.

3. Proposed Method

In this paper, we propose a taskonomy-driven PAD method (**TOD**) to improve the generalization of PAD model by adopting auxiliary information from face-related tasks. As shown in Figure 2, the proposed **TOD** method consists of two branches, including a CNN based PA detector and an auxiliary branch. The CNN based PA detector disentangles disparities between bona fides and PAs by directly extracting features from image space. The auxiliary branch aims to extract PAD-specific features from a model trained by face-related tasks. In such auxiliary branch, we first hierarchically extract the task-specific features from multiple layers of the trained model. Then, we design a Cross-modal Adapter based on GNN to adapt the features to PAD. The features from both branches are fused comprehensively for the final PAD. In the following sections, we will present the detailed discussion on the proposed method.

3.1. Task Specific Features from Face-related Tasks

As some face-related tasks are trained from huge amount of faces, features extracted by such trained networks have better generalization capability. The hypothesis of this work is that face-related tasks share some common patterns in face feature learning. For example, expression recognition requires the model to localize the action unit of the face, which also serves as a potential feature for PAD. Through transferring the knowledge contained in trained tasks, the dependence of PA detector on a large training data can be reduced. Hence features from face-related tasks not only perform strong generalization in the trained task, but can also boost the performance of PA detector. Let x refers to a face in training set, and denote $FR(\cdot)$ as the network trained by face-related tasks, e.g. face recognition, expression recognition and attribute editing. As shown in Fig. 2, $FR(\cdot)$ embeds x to a task specific feature $T = \{t_i | i \in [1, n]\}$, where t_i refers to the feature map extracted from the i-th layer. As a multi-level representation of x, the features from different layer represent different properties of the face. To prevent from the loss of information, the proposed method regards such non-structure feature map as the input of the auxiliary branch.

3.2. Cross-Modal Representation Using GAT

As $FR(\cdot)$ is trained by the other face tasks, the extracted features t_i might contains information unrelated to the face PAD task. To alleviate the potential negative influence of the irrelevant information, we propose a cross-modal adapter to re-map them for PAD. Considering that task specific features are non-structural, a Graph Neural

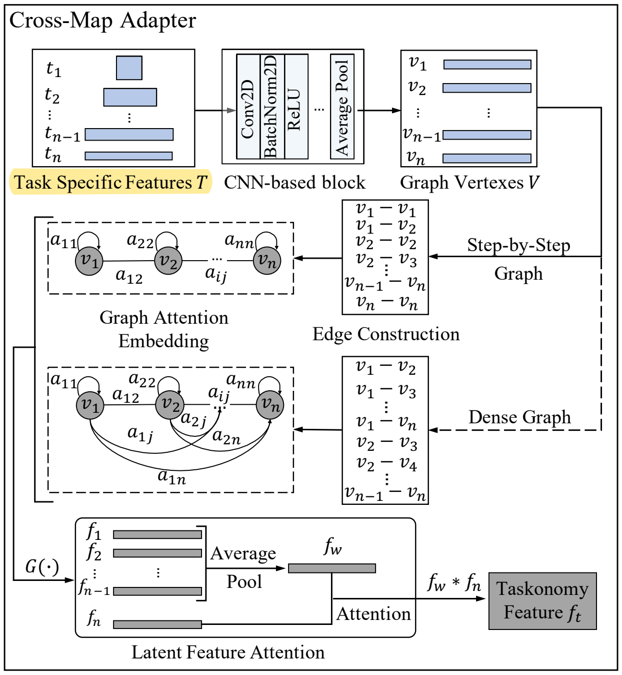


Figure 3. The pipeline of Cross-Modal Adapter. To transform the task specific features $t_i \in T$ to graph vertexes, t_i are reconstructed to one-dimension vectors $v_i \in V$ using a CNN-based block. We design two different graphs, including Step-by-Step and Dense Graph. In both graphs, attention mechanisms are used to specify the connection strength between different v_i. Through graph embedding $G(\cdot)$, v_i can be re-mapped to f_i. Then, the latent feature f_i is used to compute latent feature attention f_w by an average pool operation. The final taskonomy feature f_t is obtained by $f_w * f_n$.

Network (GNN), denoted as G(V, E), is employed to process T. $v_i \in V$ denotes vertex feature of graph. E is the edge matrix of graph to connect neighboring vertexes given by:

$$E = \begin{cases} e_{ij} = 1 & v_i \rightarrow v_j \\ e_{ij} = 0 & v_i \overset{\times}{\rightarrow} v_j \end{cases} \quad (1)$$

Given two graph vertexes v_i and v_j, $e_{ij} = 1$ presents an undirected edge existing between them. To construct neighboring v_i properly in E, the relationship among v_i is needed to be exploited. Since t_i is extracted by $FR(\cdot)$ step by step, T can perform as a sequence. Based on such observation, we propose two potential graph structures, including Step-by-Step Graph and Dense Graph, to investigate the reasonable utilization of T.

However, matrix E can only reflect the connection of each v_i. The importance of different v_i is unknown. Thus, we adopt an attention mechanism in GNN (*i.e.* GAT) embedding to find out the contribution of neighboring vertex to the central vertex. More important vertex features will obtain larger weights. Formally, the relative weights of the connected graph vertexes can be computed by:

$$A = E * (q_1WV + (q_2WV)^T) \tag{2}$$

where $Q = \{q_1, q_2\}$ is set as the shared attention parameter. A shared learnable weight matrix W is set to achieve graph convolutional operation, which can map each vertex v_i to a high-level feature. A stands for the attention matrix of graph vertexes V. Then, we use softmax to normalize $a_{ij} \in A$ by following the rule of connection:

$$A_s(i,j) = \frac{exp(a_{ij})}{\sum_{k\in[1,n]} exp(a_{ik}))} \tag{3}$$

Given two connected points v_i and v_j , $A_s(i,j)$ and $A_s(j,i)$ measures the connection strength coefficient of them. For one-layer attention, the vertex features with attention weights can be obtained:

$$V^{'} = A_sWV \tag{4}$$

As shown in Fig. 3, task specific features t_i are firstly embedded into graph vertex features v_i by a CNN-based block. Then, for Step-by-Step Graph, two graph vertexes v_i and v_{i+1} are sequentially connected by a single edge. Different graph vertexes are fully connected in Dense Graph. Through the multi-layer graph embedding $G(V^{'}, E)$, the transformed features f_i are obtained. We regard latent features as the attention weights to strengthen the representation of f_n. Through an average pooling operation along with $f_i, i \in [1, n-1]$, the latent feature attention f_w can be computed. The final taskonomy feature f_t is represented by $f_w * f_n$.

3.3. TOD based Presentation Attack Detection

To adopt the re-mapped taskonomy feature in face PAD, the proposed method introduces a CNN based PA detector $D(\cdot)$ to learn the PAD feature f_p. Then, a hierarchical feature f_h is derived by concatenating PAD feature f_p and taskonomy feature f_t. Through f_h, classifier $C(\cdot)$ can effectively distinguish bona fides with PAs . In the training process, $D(\cdot)$, $G(\cdot)$ and $C(\cdot)$ are trained by a cross entropy as follows:

$$\mathcal{L}_{x_j \in \mathcal{X}_T}(x_j, y_j^{'}) = -\frac{1}{N}\sum_{j=1}^{N}[y_j log(y_j^{'}) + (1-y_j) log(1-y_j^{'})] \tag{5}$$

where $(x_j, y_j), j \in [1, N]$ are the paired samples from training set $\mathcal{X}_T$, and $y_j^{'}$ is the prediction result of $C(\cdot)$. For clarity, the proposed method is summarized in Algorithm 1.

4. Experimental Results and Analysis

In this section, we evaluate the performance of the proposed method by carrying experiments on the publicly-available datasets [3, 7, 15, 16], Firstly, the datasets and the corresponding implementation details are introduced. Then, we validate the effectiveness of the proposed method through analyzing the influences of each network component to PAD performance. Finally, to prove the superiority of our method, we compare the PAD performance of the proposed method with the state-of-the-art methods.

Algorithm 1 Presentation Attack Detection Using TOD

```
Input:
  Training Set $\mathcal{X}_T$; CNN based PA detector $D(\cdot)$;
  Face-related network $FR(\cdot)$; Classifier $C(\cdot)$;
  Graph embedding $G(\cdot)$;
Output:
  Trained $D(\cdot)$, $G(\cdot)$ and $C(\cdot)$;
Fixed parameters of Trained $FR(\cdot)$;
for $x_j$ in $X_T$ do
    Extract PAD feature $f_p$ through $D(x_j)$;
    Derive task specific features $t_i \in T$ from $FR(x_j)$;
    Transform $t_i$ to vector $v_i \in V$ as graph vertexes;
    Construct edge matrix $E$;
    Derive vertex features $V^{'}$ with attention weighs;
    Extract transformed features $f_i$ through $G(V^{'}, E)$;
    Obtain attention weights $f_w$ from $f_i, i \in [1, n-1]$;
    Calculate taskonomy feature $f_t$ by $f_w * f_n$;
    Derive hierarchical feature $f_h$ from $f_p$ and $f_t$;
    Predict the PAD result by $C(f_h)$;
    Update $D(\cdot)$, $G(\cdot)$ and $C(\cdot)$by minimizing Eq. (5);
end for
Return $D(\cdot)$, $G(\cdot)$ and $C(\cdot)$;
```

Making your pseudo map generation as the algorithm block here.

4.1. Datasets and Implementation Details

We use four public face anti-spoofing datasets, including OULU-NPU [3] (denoted as O), CASIA-FASD [15] (denoted as C), Idiap Replay-Attack [16] (denoted as I) and MSU-MFSD [7] (denoted as M) to evaluate the effectiveness of our method. Existing methods were evaluated on the protocol [12], denoted as Protocol-I. In this protocol, three of datasets are used as training set and the remaining one is used for test. However, in the reality, there are much more unseen PAs than the known ones in the training set. Using $3/4$ datasets to train model is not strict to the real application scenario. Thus, we design a different cross-dataset protocol (Protocol-II) to evaluate the generalization ability of our method. In detail, we only use two datasets from [O, M, C, I] to train model and the remaining two datasets to test. Due to the number of samples varies greatly among each dataset, some data divisions will be unreasonable for model training. To make the number of training set and test set as close as possible, we only divide the datasets into two groups, *i.e.* [O, M] and [C, I]. To reduce the influence caused by the background, resolution, and illustration, MTCNN algorithm [31] is used for

Table 1. Performance of the Proposed Method with or without Taskonomy Features Obtained from Three Different Tasks.

	[O.M] to [C,I]			[C,I] to [O,M]			**Mean ± s.d.**		
	HTER(%)	AUC(%)	TDR(%)	HTER(%)	AUC(%)	TDR(%)	HTER(%)	AUC(%)	TDR(%)
Baseline	25.65	79.14	4.07	28.14	79.05	18.66	26.90 ± 1.76	79.10 ± 0.06	11.37 ± 10.32
Baseline w/ *F.R.*	19.81	87.77	37.61	16.47	90.68	37.19	18.14 ± 2.36	89.23 ± 2.06	37.40 ± 0.300
Baseline w/ *F.E.*	17.93	85.97	9.1	16.62	91.78	44.34	17.28 ± 0.93	88.88 ± 4.11	26.72 ± 24.92
Baseline w/ *F.A.*	**16.98**	**90.66**	**41.68**	**13.18**	**94.36**	**56.30**	**15.08 ± 2.69**	**92.51 ± 2.62**	**48.99 ± 10.34**

Table 2. Performance of the Proposed Method Using Different models in Cross-Modal Adapter for Three Taskonomy Features.

Taskonomy Features	**Cross-Modal Adapters**	[O.M] to [C,I]			[C,I] to [O,M]			**Mean ± s.d.**		
		HTER(%)	AUC(%)	TDR(%)	HTER(%)	AUC(%)	TDR(%)	HTER(%)	AUC(%)	TDR(%)
×	×	25.65	79.14	4.07	28.14	79.05	18.66	26.90 ± 1.76	79.10 ± 0.06	11.37 ± 10.32
F.R.	CNN	19.89	87.56	21.29	16.70	91.59	47.40	18.30 ± 2.26	89.58 ± 2.85	34.35 ± 18.46
	Transformer	19.53	86.18	18.31	17.72	90.53	31.86	18.63 ± 1.28	88.36 ± 3.08	25.09 ± 9.580
	GAT	18.17	87.37	21.48	16.47	90.68	37.19	17.32 ± 1.20	89.03 ± 2.34	29.34 ± 11.11
F.E.	CNN	21.58	86.47	34.05	17.94	89.93	42.14	19.76 ± 2.57	88.20 ± 2.45	38.10 ± 5.720
	Transformer	19.23	85.40	8.81	16.76	91.31	43.84	18.00 ± 1.75	88.36 ± 4.18	26.33 ± 24.77
	GAT	17.93	85.97	9.1	16.62	91.78	44.34	17.28 ± 0.93	88.88 ± 4.11	26.72 ± 24.92
F.A.	CNN	18.05	86.34	10.07	16.55	90.50	39.84	17.30 ± 1.06	88.42 ± 2.94	24.96 ± 21.05
	Transformer	20.59	87.72	34.51	20.58	87.57	32.13	20.59 ± 0.01	87.65 ± 0.11	33.32 ± 1.680
	GAT	**16.98**	**90.66**	**41.68**	**13.18**	**94.36**	**56.30**	**15.08 ± 2.69**	**92.51 ± 2.62**	**48.99 ± 10.34**

face detection and alignment. All the detected faces are resized to (256, 256). ResNet18 [32] is fine-tuned as the CNN based PA detector. Three network trained through the face-related tasks are used to obtain task specific features. ResNet18 trained by face recognition task [23] and face expression recognition [24] is set as the feature extractor. For face attribute editing task, the trained discriminator of styleGAN [25] is set to extract task specific features. In the training process, the parameters of networks with face-related tasks are fixed. In the graph embedding, a two-layer GAT with two-head attention mechanisms is adopted. To evaluate the cross-modal adapter of our method, besides GAT, we adopt other deep learning methods, including ResNet18 and transformer [33], as the competing methods.

In summary, we train PA detector, classifier and cross-modal adapter by Adam with 1e-4 learning rate, 0.9 momentum and 5e-5 weight decay. Batch size for training is 32. To validate the superiority of our TOD method, the state-of-the-art PAD methods, including DeepPixBiS [11], SSDG-R [12], CDC [18], and IF-OM [34] are conducted in this paper. Following the work of [34], We use Half Total Error Rate (HTER), Area Under Curve (AUC) and True Detection Rate (TDR) @ False Detection Rate (FDR)=1% to evaluate the performance of PAD. This paper adopts the public platform pytorch for all experiments using a work station with CPUs of 2.8GHz, RAM of 512GB and GPUs of NVIDIA Tesla V100.

4.2. Effectiveness Analysis of the Proposed Method

We perform the ablation study to quantify the influence of each component in our model for face PAD. First, to evaluate the effectiveness of taskonomy features, we test the performance of PAD with or without the taskonomy features. Table 1 shows the results carried on the cross-dataset protocol. The baseline is set as the ResNet18 model pretrained from ImageNet. We use three face-related tasks, including face recognition, face expression recognition, and face attribute editing to extract task specific features. Then, by a step-by-step GAT, we can respectively obtain three different taskonomy features ($F.R.$, $F.E.$, $F.A.$). Compared

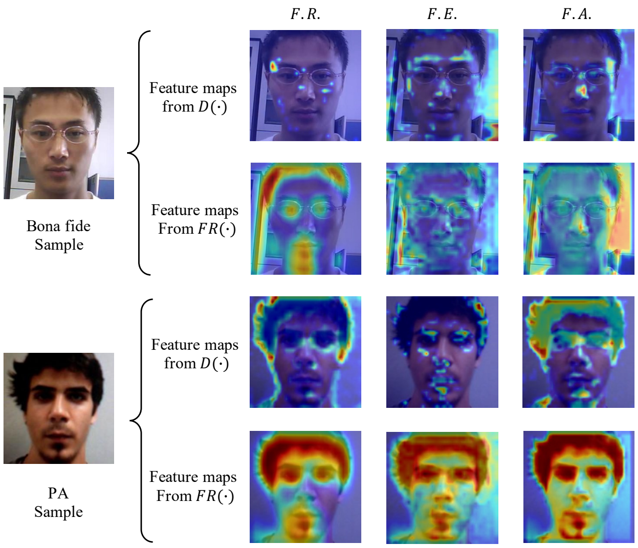
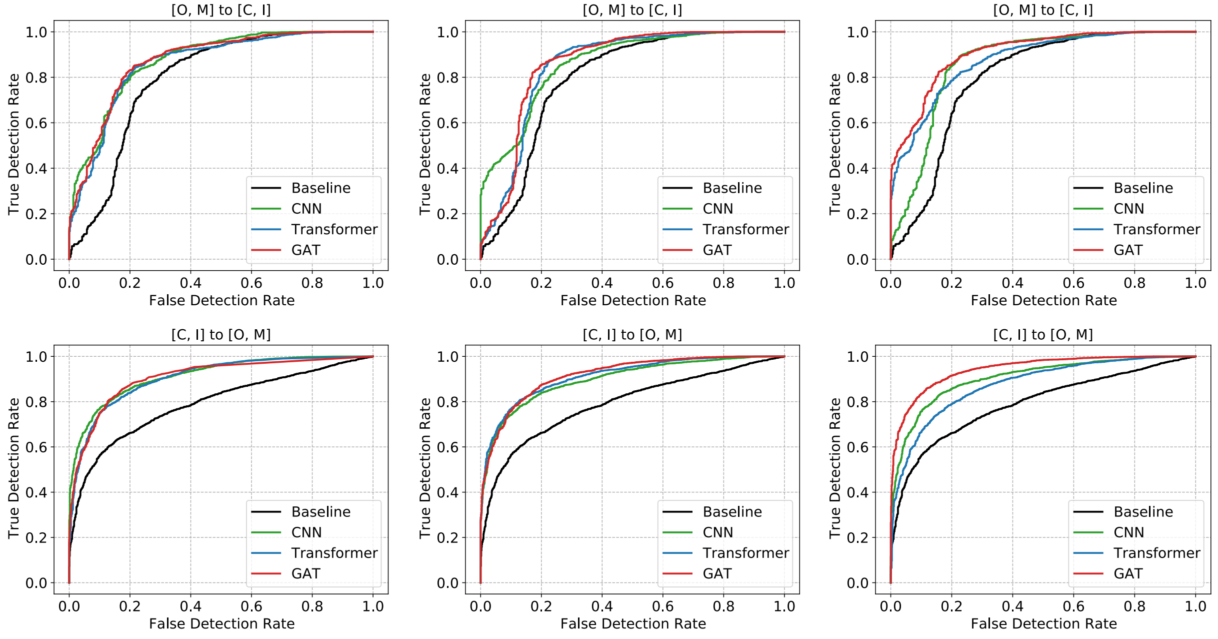


Figure 4. The visualization on CASIA-FASD using Grad-CAM. The first row for each sample shows the discriminative regions obtained from CNN-based PA detector $D(\cdot)$ and the second row for each sample illustrate the region localizations extracted from network of Face-Related Tasks $FR(\cdot)$.



(a) $F.R.$ (b) $F.E.$ (c) $F.A.$

Figure 5. ROC curves for the ablation study when using different models in Cross-Modal Adapter. The experiments are under the cross-dataset setting (Protocol-II) and adopting three different taskonomy features, which are (a) $F.R.$, (b) $F.E.$ and (c) $F.A.$. Baseline (black line) in the figure represents the model without using Cross-Modal Adapter.

with the baseline, all three taskonomy features can improve the PAD performance. Specifically, $F.A.$ feature adapted from face attribute editing task improves the HTER of baseline from 26.90% to 15.08%. This indicates that taskonomy features are useful to face PAD. As experiments are carried on the cross-dataset protocol, it also indicates that taskonomy features can improve generalization ability of face PAD.

To further verify the effectiveness of taskonomy features, we adopt Grad-CAM [35] to visualize the discriminative regions from feature maps in our proposed TOD model. We compare the visualization results with three taskonomy features. Cross-Modal Adapters are set as the Step-by-Step Graphs. As shown in Fig. 4, when using taskonomy features, the model can find discriminative features for both bona fide and PA samples. In the visualization region obtained form $FR(\cdot)$, the visualization shows that the hair, eyes, nose and mouth are important to distinguish live faces and spoofs. This further indicates the effectiveness of the taskonomy features. Typically, comparing with the visualization of $F.R.$ and $F.E.$ taskonomy features, $F.A.$ features can provide more effective region of face attributes.

For each task specific feature v_i, we further compare two other different deep learning models with GAT, *i.e.* CNN based model and transformer model, to justify the effectiveness of the Cross-Modal Adapter. In CNN based model, we use the same CNN-based block and latent feature attention in Fig. 3 to obtain the taskonomy feature. In transformer based model, each v_i is transformed to vector adopting CNN-based block in Fig. 3 and encoded with position encoding module in [33]. Then, We adopt six-layer transformer encoders with eight-head-attention to obtain the taskonomy feature. As CNN model and transformer model are sequential models, we only use the Step-by-Step Graph to ensure the fairness of the comparison. The PAD results in Table 2 show that the performance of the proposed method with different deep learning models in Cross-Modal Adapter and taskonomy features is better than baseline.

Corresponding to Table 2, Fig. 5 presents the ROC (Receiver Operating Characteristic) curves of baseline and three taskonomy features when using different model in Cross-Modal Adapters. It can be seen that, for all taskonomy features, Cross-Modal Adapter using GAT model (with red lines in Fig. 5) achieves a higher performance than CNN model and transformer model. This indicates that GAT model is more suitable for the Cross-Model Adapter. In particular, taskonomy feature $F.A.$ re-mapped from GAT can obtain the best results in both cross-dataset protocols, and achieve an average HTER of 15.08% and AUC of 92.51%. These results can verify the contribution of Cross-Modal Adapter to improve the PAD performance and generalization.

Table 3. Performance Comparison between the Proposed Method and the State-Of-The-Art Methods under the Cross-Dataset Setting. (Protocol-I).

Method	[O, C, I] to M		[O, M, I] to C		[O, C, M] to I		[I, C, M] to O		**Mean ± S.d.**	
	HTER (%)	AUC(%)	HTER (%)	AUC(%)	HTER (%)	AUC(%)	HTER (%)	AUC(%)	HTER (%)	AUC(%)
MS-LBP [36]	29.76	78.50	54.28	44.98	50.30	51.64	50.29	49.31	46.16 ± 9.610	56.11 ± 13.15
Binary CNN [10]	29.25	82.87	34.88	71.94	34.47	65.88	29.61	77.54	32.05 ± 2.630	74.56 ± 6.330
IDA [7]	66.67	27.86	55.17	39.05	28.35	78.25	54.20	44.59	51.10 ± 14.02	47.44 ± 18.78
Color Texture [37]	28.09	78.47	30.58	76.89	40.40	62.78	63.59	32.71	40.67 ± 14.01	62.71 ± 18.37
LBP-TOP [38]	36.90	70.80	33.52	73.15	29.14	71.69	30.17	77.61	32.43 ± 3.050	73.31 ± 2.620
Auxiliary [19]	-	-	28.40	-	27.60	-	-	-	-	-
MADDG [21]	17.69	88.06	24.50	84.51	22.19	84.99	27.89	80.02	23.07 ± 3.710	84.40 ± 2.870
SSDG-R [12]	7.38	97.17	10.44	95.94	11.71	**96.59**	15.61	91.54	11.29 ± 2.950	95.31 ± 2.220
IF-OM [34]	7.14	97.09	15.33	91.41	14.03	94.30	16.68	91.85	13.30 ± 3.680	93.66 ± 2.260
Baseline	13.10	92.76	16.44	91.25	24.58	79.50	22.31	85.65	19.11 ± 5.270	87.29 ± 6.030
Ours: TOD	**5.71**	**97.21**	**10.33**	**96.73**	**11.37**	94.79	**13.55**	**94.64**	**10.24 ± 3.300**	**95.84 ± 1.320**

Table 4. Performance Comparison between the Proposed Method and the State-Of-The-Art Methods under the Cross-Dataset Setting. (Protocol-II).

	[O, M] to [C,I]			[C, I] to [O, M]			**Mean ± S.d.**		
	HTER(%)	AUC(%)	TDR(%)	HTER(%)	AUC(%)	TDR(%)	HTER(%)	AUC(%)	TDR(%)
CDC [18]	28.94	78.96	13.93	23.30	83.42	25.83	26.12 ± 3.99	81.19 ± 3.15	19.88 ± 8.41
DeepPixBiS [11]	22.93	79.13	0.00	22.45	85.70	24.37	22.69 ± 0.34	82.42 ± 4.65	12.19 ± 17.23
SSDG-R [12]	20.92	88.07	9.72	22.57	85.61	15.95	21.75 ± 1.17	86.84 ± 1.74	12.84 ± 4.41
IF-OM [34]	18.96	89.48	30.48	18.60	89.76	30.30	18.78 ± 0.25	89.62 ± 0.20	30.39 ± 0.13
Baseline	25.65	79.14	4.07	28.14	79.05	18.66	26.90 ± 1.76	79.10 ± 0.06	11.37 ± 10.32
Ours: TOD w/ Dense Graph	18.78	87.99	12.07	16.3	92.32	47.27	17.54± 1.75	90.16 ± 3.06	29.67 ±24.89
Ours: TOD w/ Step-by-Step Graph	**16.98**	**90.66**	**41.68**	**13.18**	**94.36**	**56.30**	**15.08 ± 2.69**	**92.51 ± 2.62**	**48.99 ± 10.34**

4.3. Comparison with other Methods

To further verify the effectiveness of the proposed method, we compare it with the state-of-the-art methods in two protocols. Table 3 lists the comparison results in Protocol-I. Here, we give our results using the best model (*i.e.* adopting Step-by-Step Graph as the cross-modal Adapter of the $F.A.$ feature). It can be seen that, the proposed TOD method outperforms the state-of-the-art methods *e.g.* IF-OM and SSDG-R by the average HTER and AUC. Specifically, in experiment [I, C, M] to O, our method can outperform both SSDG-R and IF-OM by average 2.60% HTER.

Moreover, in more challenge Protocol-II (two datasets as training set and other two datasets as test set), we can obtain good results with both adapters (Dense Graph and Step-by-Step Graph) using the $F.A.$ feature. As shown in Table 4, the proposed method based on Step-by-Step Graph can outperform other methods by a large margin. Compared with CDC [18], our method can improve the average HTER of PAD from 26.12% to 15.08% and AUC from 81.18% to 92.51%. These results indicate that the proposed method can generalize better than other PAD methods in both protocols, which further prove the superiority of our method.

5. Conclusion

Existing face presentation attack detection methods cannot generalize well to unseen PAs, due to the highly dependence on the limited datasets. In this paper, to improve generalization ability of face PAD, we proposed a taskonomy-driven face PAD method. By designing a Cross-Modal Adapter, features from other face-related tasks can re-map to more effective anti-spoofing features for PAD. Experimental results have shown the effectiveness of the proposed method. Compared with the state-of-the-art methods in existing dataset partition (*i.e.* Protocol-I), we can improve average HTER from 11.29% to 10.24% and AUC from 95.31% to 95.84%. Furthermore, when the dataset partition becomes more challenging (*i.e.* Protocol-II where more PAs are unseen to the model), our method largely improve the average HTER to 15.08%, which demonstrates the strong generalization ability of our method to handle unpredictable PAs.

References

[1] Raghavendra Ramachandra and Christoph Busch. Presentation attack detection methods for face recognition systems: A comprehensive survey. *ACM Computing Surveys (CSUR)*, 50(1):1–37, 2017. 1

[2] Shan Jia, Guodong Guo, and Zhengquan Xu. A survey on 3d mask presentation attack detection and countermeasures. *Pattern Recognition*, 98:107032, 2020. 1

[3] Zinelabinde Boulkenafet, Jukka Komulainen, Lei Li, Xiaoyi Feng, and Abdenour Hadid. Oulu-npu: A mobile face presentation attack database with real-world variations. In *2017 12th IEEE international conference on automatic face & gesture recognition (FG 2017)*, pages 612–618. IEEE, 2017. 1, 2, 5

[4] Guillaume Heusch, Anjith George, David Geissbühler, Zohreh Mostaani, and Sébastien Marcel. Deep models and shortwave infrared information to detect face presentation at-

tacks. *IEEE Transactions on Biometrics, Behavior, and Identity Science*, 2(4):399–409, 2020. 1

[5] Ramachandra Raghavendra, Kiran B Raja, and Christoph Busch. Presentation attack detection for face recognition using light field camera. *TIP*, 24(3):1060–1075, 2015. 1

[6] Tiago de Freitas Pereira, André Anjos, José Mario De Martino, and Sébastien Marcel. Can face anti-spoofing countermeasures work in a real world scenario? In *2013 international conference on biometrics (ICB)*, pages 1–8. IEEE, 2013. 1, 2

[7] Di Wen, Hu Han, and Anil K Jain. Face spoof detection with image distortion analysis. *IEEE Transactions on Information Forensics and Security*, 10(4):746–761, 2015. 1, 2, 5, 8

[8] Keyurkumar Patel, Hu Han, and Anil K Jain. Secure face unlock: Spoof detection on smartphones. *IEEE transactions on information forensics and security*, 11(10):2268–2283, 2016. 1, 2

[9] Zinelabidine Boulkenafet, Jukka Komulainen, and Abdenour Hadid. Face antispoofing using speeded-up robust features and fisher vector encoding. *IEEE Signal Processing Letters*, 24(2):141–145, 2016. 1, 2

[10] Jianwei Yang, Zhen Lei, and Stan Z Li. Learn convolutional neural network for face anti-spoofing. *arXiv preprint arXiv:1408.5601*, 2014. 1, 2, 8

[11] Anjith George and Sébastien Marcel. Deep pixel-wise binary supervision for face presentation attack detection. In *2019 International Conference on Biometrics (ICB)*, pages 1–8. IEEE, 2019. 1, 6, 8

[12] Yunpei Jia, Jie Zhang, Shiguang Shan, and Xilin Chen. Single-side domain generalization for face anti-spoofing. In *CVPR*, pages 8484–8493, 2020. 1, 5, 6, 8

[13] Yahang Wang, Xiaoning Song, Tianyang Xu, Zhenhua Feng, and Xiao-Jun Wu. From rgb to depth: Domain transfer network for face anti-spoofing. *IEEE Transactions on Information Forensics and Security*, 16:4280–4290, 2021. 1

[14] Anjith George and Sébastien Marcel. Cross modal focal loss for rgbd face anti-spoofing. In *CVPR*, pages 7882–7891, 2021. 1, 2

[15] Zhiwei Zhang, Junjie Yan, Sifei Liu, Zhen Lei, Dong Yi, and Stan Z Li. A face antispoofing database with diverse attacks. In *2012 5th IAPR international conference on Biometrics (ICB)*, pages 26–31. IEEE, 2012. 2, 5

[16] Ivana Chingovska, André Anjos, and Sébastien Marcel. On the effectiveness of local binary patterns in face anti-spoofing. In *2012 BIOSIG-proceedings of the international conference of biometrics special interest group (BIOSIG)*, pages 1–7. IEEE, 2012. 2, 5

[17] Son Minh Nguyen, Linh Duy Tran, and Masayuki Arai. Attended-auxiliary supervision representation for face anti-spoofing. In *ACCV*, 2020. 2

[18] Zitong Yu, Chenxu Zhao, Zezheng Wang, Yunxiao Qin, Zhuo Su, Xiaobai Li, Feng Zhou, and Guoying Zhao. Searching central difference convolutional networks for face anti-spoofing. In *CVPR*, pages 5295–5305, 2020. 2, 6, 8

[19] Yaojie Liu, Amin Jourabloo, and Xiaoming Liu. Learning deep models for face anti-spoofing: Binary or auxiliary supervision. In *CVPR*, pages 389–398, 2018. 2, 8

[20] Anjith George and Sébastien Marcel. Learning one class representations for face presentation attack detection using multi-channel convolutional neural networks. *IEEE Transactions on Information Forensics and Security*, 16:361–375, 2020. 2

[21] Rui Shao, Xiangyuan Lan, Jiawei Li, and Pong C Yuen. Multi-adversarial discriminative deep domain generalization for face presentation attack detection. In *CVPR*, pages 10023–10031, 2019. 2, 8

[22] Guoqing Wang, Hu Han, Shiguang Shan, and Xilin Chen. Cross-domain face presentation attack detection via multi-domain disentangled representation learning. In *CVPR*, pages 6678–6687, 2020. 2

[23] Jiankang Deng, Jia Guo, Niannan Xue, and Stefanos Zafeiriou. Arcface: Additive angular margin loss for deep face recognition. In *CVPR*, pages 4690–4699, 2019. 3, 6

[24] Kai Wang, Xiaojiang Peng, Jianfei Yang, Shijian Lu, and Yu Qiao. Suppressing uncertainties for large-scale facial expression recognition. In *CVPR*, pages 6897–6906, 2020. 3, 6

[25] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *CVPR*, pages 4401–4410, 2019. 3, 6

[26] Qi Wu, Damien Teney, Peng Wang, Chunhua Shen, Anthony Dick, and Anton van den Hengel. Visual question answering: A survey of methods and datasets. *Computer Vision and Image Understanding*, 163:21–40, 2017. 3

[27] Ruohan Gao and Kristen Grauman. Visualvoice: Audio-visual speech separation with cross-modal consistency. In *CVPR*, pages 15495–15505, June 2021. 3

[28] Jiyoung Lee, Soo-Whan Chung, Sunok Kim, Hong-Goo Kang, and Kwanghoon Sohn. Looking into your speech: Learning cross-modal affinity for audio-visual speech separation. In *CVPR*, pages 1336–1345, June 2021. 3

[29] Amir R Zamir, Alexander Sax, William Shen, Leonidas J Guibas, Jitendra Malik, and Silvio Savarese. Taskonomy: Disentangling task transfer learning. In *CVPR*, pages 3712–3722, 2018. 3

[30] Kshitij Dwivedi and Gemma Roig. Representation similarity analysis for efficient task taxonomy & transfer learning. In *CVPR*, pages 12387–12396, 2019. 3

[31] Kaipeng Zhang, Zhanpeng Zhang, Zhifeng Li, and Yu Qiao. Joint face detection and alignment using multitask cascaded convolutional networks. *IEEE Signal Processing Letters*, 23(10):1499–1503, 2016. 5

[32] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016. 6

[33] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia

Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017. 6, 7

[34] Haozhe Liu, Zhe Kong, Raghavendra Ramachandra, Feng Liu, Linlin Shen, and Christoph Busch. Taming self-supervised learning for presentation attack detection: In-image de-folding and out-of-image de-mixing. *arXiv preprint arXiv:2109.04100*, 2021. 6, 8

[35] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *ICCV*, pages 618–626, 2017. 7

[36] Jukka Määttä, Abdenour Hadid, and Matti Pietikäinen. Face spoofing detection from single images using micro-texture analysis. In *2011 international joint conference on Biometrics (IJCB)*, pages 1–7. IEEE, 2011. 8

[37] Zinelabidine Boulkenafet, Jukka Komulainen, and Abdenour Hadid. Face spoofing detection using colour texture analysis. *IEEE Transactions on Information Forensics and Security*, 11(8):1818–1830, 2016. 8

[38] Tiago de Freitas Pereira, Jukka Komulainen, André Anjos, José Mario De Martino, Abdenour Hadid, Matti Pietikäinen, and Sébastien Marcel. Face liveness detection using dynamic texture. *EURASIP Journal on Image and Video Processing*, 2014(1):1–15, 2014. 8