Federated Face Presentation Attack Detection

Rui Shao[1]   Pramuditha Perera[2]   Pong C. Yuen[1]   Vishal M. Patel[2]

[1]Department of Computer Science, Hong Kong Baptist University
[2]Department of Electrical and Computer Engineering, Johns Hopkins University

Abstract

Face presentation attack detection (fPAD) plays a critical role in the modern face recognition pipeline. A face presentation attack detection model with good generalization can be obtained when it is trained with face images from different input distributions and different types of spoof attacks. In reality, training data (both real face images and spoof images) are not directly shared between data owners due to legal and privacy issues. In this paper, with the motivation of circumventing this challenge, we propose Federated Face Presentation Attack Detection (FedPAD) framework. FedPAD simultaneously takes advantage of rich fPAD information available at different data owners while preserving data privacy. In the proposed framework, each data owner (referred to as data centers) locally trains its own fPAD model. A server learns a global fPAD model by iteratively aggregating model updates from all data centers without accessing private data in each of them. Once the learned global model converges, it is used for fPAD inference. We introduce the experimental setting to evaluate the proposed FedPAD framework and carry out extensive experiments to provide various insights about federated learning for fPAD.

1. Introduction

Recent advances in face recognition methods have prompted many real-world applications, such as automated teller machines (ATMs), mobile devices, and entrance guard systems, to deploy this technique as an authentication method. Wide usage of this technology is due to both high accuracy and convenience it provides. However, many recent works [13, 3, 23, 19, 11, 18, 20, 21] have found that this technique is vulnerable to various face presentation attacks such as print attacks, video-replay attacks [2, 24, 4, 23, 11] and 3D mask attacks [9, 10]. Therefore, developing face presentation attack detection (fPAD) methods that make current face recognition systems robust to face presentation attacks has become a topic of interest in the biometrics community.
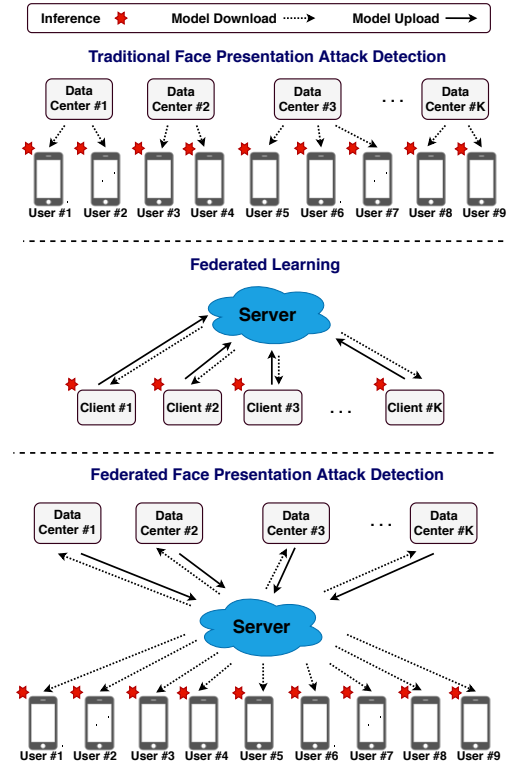

Figure 1. Comparison between fPAD (top), traditional federated learning (middle) and the proposed FedPAD (bottom). FedPAD can be a regarded as a special case of traditional federated learning.

In this paper, we consider the deployment of a fPAD system in a real-world scenario. We identify two types of stakeholders in this scenario – *data centers* and *users*. *Data centers* are entities that design and collect fPAD datasets and propose fPAD solutions. Typically *data centers* include research institutions and companies that carry out the research and development of fPAD. These entities have access to both *real data* and *spoof data* and therefore are able to train fPAD models. Different *data centers* may contain images of different identities and different types of *spoof data*. However, each *data center* has limited data availability. Real face images are obtained from a small set of iden-

tities and spoof attacks are likely to be from a few known types of attacks. Therefore, these fPAD models have poor generalization ability [18, 21] and are likely to be vulnerable against attacks unseen during training.

On the other hand, *users* are individuals or entities that make use of fPAD solutions. For example, when a fPAD algorithm is introduced in mobile devices, mobile device customers are identified as *users* of the fPAD system. *Users* have access only to *real data* collected from local devices. Due to the absence of *spoof data*, they cannot locally train fPAD models. Therefore, each *user* relies on a model developed by a *data center* for fPAD as shown in Figure 1 (top). Since *data center* models lack generalization ability, inferencing with these models are likely to result in erroneous predictions.

It has been shown that utilizing *real data* from different input distributions and *spoof data* from different types of spoof attacks through domain generalization and meta-learning techniques can significantly improve the generalization ability of fPAD models [18, 21]. Therefore, the performance of fPAD models, shown in Figure 1 (top), can be improved if data from all *data centers* can be exploited collaboratively. In reality, due to data sharing agreements and privacy policies, *data centers* are not allowed to share collected fPAD data with each other. For example, when a *data center* collects face images from individuals using a social media platform, it is agreed not to share collected data with third parties.

In this paper, we present a framework called Federated Face Presentation Attack Detection (FedPAD) based on the principles of Federated Learning (FL) targeting fPAD. The proposed method exploits information across all *data centers* while preserving data privacy. In the proposed framework, models trained at *data centers* are shared and aggregated while training images are kept private in their respective *data centers*, thereby preserving privacy.

Federate learning is a distributed and privacy preserving machine learning technique [14, 8, 22, 17, 15]. FL training paradigm defines two types of roles named *server* and *client*. *Clients* contain training data and the capacity to train a model. As shown in Fig. 1 (middle), each client trains its own model locally and uploads them to the *server* at the end of each training iteration. *Server* aggregates local updates and produces a global model. This global model is then shared with all clients which will be used in their subsequent training iteration. This process is continued until the global model is converged. During the training process, data of each client is kept private. Collaborative FL training allows the global model to exploit rich local clients information while preserving data privacy.

In the context of FedPAD, both *data centers* and *users* can be identified as clients. However, roles of *data centers* and *users* are different from conventional clients found in FL. In FL, all *clients* train models and carry out inference locally. In contrast, in FedPAD, only *data centers* carry out local model training. *Data centers* share their models with the *server* and download the global model during training. On the other hand, *users* download the global model at the end of the training procedure and only carry out inference as shown in Figure 1 (bottom).

Main contributions of our paper can be summarized as follows:

1. This paper is the first to study the federated learning technique for the task of fPAD. We propose the Federated Face Presentation Attack Detection (FedPAD) framework to develop the robust and generalized fPAD model in a data privacy preserving way.

2. An experimental setting is defined for the FedPAD framework. Extensive experiments are carried out to show the effectiveness of the proposed framework. Various issues and insights about federated learning for fPAD are discussed and provided.

2. Related Work

2.1. Face Presentation Attack Detection

Current fPAD methods can be categorized under single-domain and multi-domain approaches. Single-domain approach focuses on extracting discriminative cues between real and spoof samples from a single dataset, which can be further divided into appearance-based methods and temporal-based methods. Appearance-based methods focus on extracting various discriminative appearance cues for detecting face presentation attacks. Multi-scale LBP [13] and color textures [3] methods are two texture-based methods that extract various LBP descriptors in various color spaces for the differentiation between real/spoof samples. Image distortion analysis [23] aims to detect the surface distortions as the discriminative cue.

On the other hand, temporal-based methods extract different discriminative temporal cues through multiple frames between real and spoof samples. Various dynamic textures are exploited in [16, 20, 19] to extract discriminative facial motions. rPPG signals are exploited by Liu *et al.* [10, 9] to capture discriminative heartbeat information from real and spoof videos. [11] learns a CNN-RNN model to estimate different face depth and rPPG signals between the real and spoof samples.

Various fPAD datasets are introduced recently that explore different characteristics and scenarios of face presentation attacks. Multi-domain approach is proposed in order to improve the generalization ability of the fPAD model to unseen attacks. Recent work [18] casts fPAD as a domain generalization problem and proposes a multi-adversarial discriminative deep domain generalization framework to search generalized differentiation cues in a shared and dis-
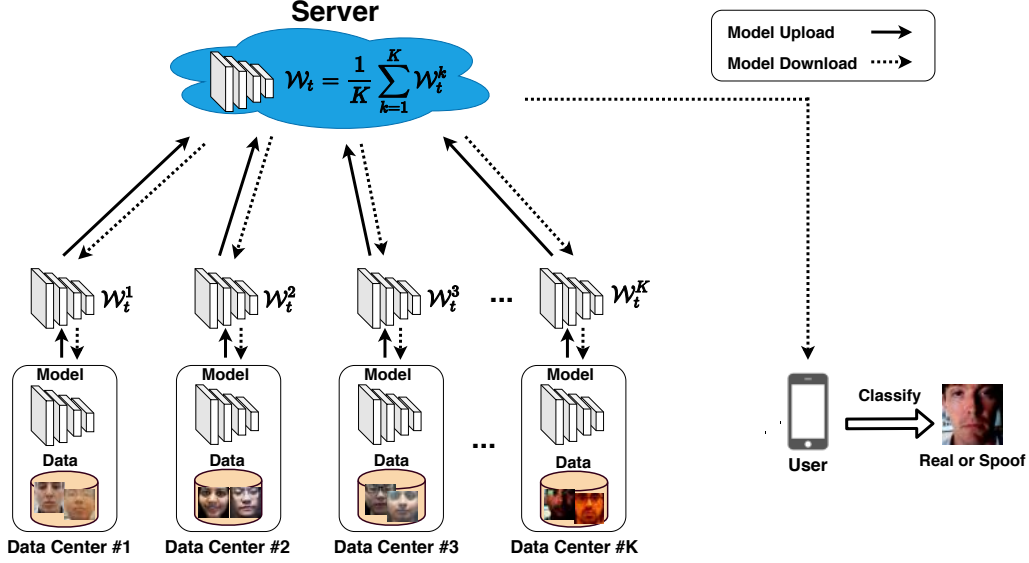
Figure 2. Overview of the proposed FedPAD framework. Trough several rounds of communication between *data centers* and *server*, the collaborated trained global fPAD model parameterized by $\mathcal{W}_t$ can be obtained in a data privacy preserving way. Users can download this global model from the server to their device to detect various face presentation attacks.

criminative feature space among multiple fPAD datasets. [12] treats fPAD as a zero-shot problem and proposes a Deep Tree Network to partition the spoof samples into multiple sub-groups of attacks. [21] addresses fPAD with a meta-learning framework and enables the model learn to generalize well through simulated train-test splits among multiple datasets. These multi-domain approaches have access to data from multiple datasets or multiple spoof sub-groups that enable them to obtain generalized models. In this paper, we study the scenario in which each *data center* contains data from a single domain. Due to data privacy issues, we assume that they do not have access to data from other *data centers*. This paper aims to exploit multi-domain information in a privacy preserving manner.

2.2. Federated Learning

Federated learning is a decentralized machine learning approach that enables multiple local clients to collaboratively learn a global model with the help of a server while preserving data privacy of local clients. Federated averaging (FedAvg) [14], one of the fundamental frameworks for FL, learns a global model by averaging model parameters from local clients. FedProx [17] and Agnostic Federated Learning (AFL) [15] are two variants of FedAvg which aim to address the bias issue of the learned global model towards different clients. These two methods achieve better models by adding proximal term to the cost functions and optimizing a centralized distribution mixed with client distributions, respectively. This paper focuses on exploiting FedAvg [14] to improve the generalization ability of a fPAD

model while securing data privacy of each *data center*.

3. Proposed Method

The proposed FedPAD framework is summarized in Fig. 2 and Algorithm 1. Suppose that K *data enters* collect their own fPAD datasets designed for different characteristics and scenarios of face presentation attacks. The corresponding collected fPAD datasets are denoted as $\mathcal{D}^1, \mathcal{D}^2, ..., \mathcal{D}^K$ with data provided with image and label pairs denoted as x and y. y are ground-truth with binary class labels (y= 0/1 are the labels of spoof/real samples). Based on the collected fPAD data, each *data center* can train its own fPAD models by iteratively minimizing the cross-entropy loss as follows:

$$\mathcal{L}(\mathcal{W}^k) = \sum_{(x,y)\sim\mathcal{D}^k} y\log\mathcal{F}^k(x) + (1-y)\log(1-\mathcal{F}^k(x)), \quad (1)$$

where the fPAD model $\mathcal{F}^k$ of the k-th *data enter* is parameterized by $\mathcal{W}^k$ ($k = 1, 2, 3, ..., K$). After optimization with several local epochs via

$$\mathcal{W}^k \leftarrow \mathcal{W}^k - \eta\nabla\mathcal{L}(\mathcal{W}^k),$$

each *data enter* can obtain the trained fPAD model with the updated model parameters.

It should be noted that dataset corresponding to each *data enter* is from a specific input distribution and it only contains a finite set of known types of spoof attack data.

Algorithm 1 Federated Face Presentation Attack Detection

Require:

  Input: K Data Centers have K fPAD datasets $\mathcal{D}^1, \mathcal{D}^2, ..., \mathcal{D}^K$,

  Initialization: K Data Centers have K fPAD models $\mathcal{F}^1, \mathcal{F}^2, ..., \mathcal{F}^K$ parameterized by $\mathcal{W}_0^1, \mathcal{W}_0^2, ..., \mathcal{W}_0^K$. L is the number of local epochs. η is the learning rate. t is the federated learning rounds

  Server aggregates:

  initialize $\mathcal{W}_0$

  for each round $t = 0, 1, 2,...$ **do**

    for each data center $k = 1, 2,..., K$ **in parallel do**

      $W_t^k \leftarrow$ **DataCenterUpdate**$(k, \mathcal{W}_t)$

    end for

    $\mathcal{W}_t = \frac{1}{K} \sum\limits_{k=1}^{K} \mathcal{W}_t^k$

    Download W_t to **Data Centers**

  end for

  Users Download W_t

  DataCenterUpdate$(k, \mathcal{W})$:

  for each local epoch $i = 1, 2,..., L$ **do**

    $\mathcal{L}(\mathcal{W}^k) = \sum\limits_{(x,y)\sim\mathcal{D}^k} y \log \mathcal{F}^k(x) + (1 - y) \log(1 - \mathcal{F}^k(x))$

    $\mathcal{W}^k \leftarrow \mathcal{W}^k - \eta \nabla \mathcal{L}(\mathcal{W}^k)$

  end for

  Upload W^k to **Server**

When a model is trained using this data, it focuses on addressing the characteristics and scenarios of face presentation attacks prevalent in the corresponding dataset. However, a model trained from a specific *data center* will not generalize well to unseen face presentation attacks. It is well known fact that diverse fPAD training data contributes to a better generalized fPAD model. A straightforward solution is to collect and combine all the data from K data centers denoted as $\mathcal{D} = \{\mathcal{D}^1 \cup \mathcal{D}^2 \cup ... \cup \mathcal{D}^K\}$ to train a fPAD model. It has been shown that domain generalization and meta-learning based fPAD methods can further improve the generalization ability with the above combined multi-domain data $\mathcal{D}$ [18, 21]. However, when sharing data between different *data centers* are prohibited due to the privacy issue, this naive solution is not practical.

To circumvent this limitation and enable various data centers to collaboratively train a fPAD model, we propose the FedPAD framework. Instead of accessing private fPAD data of each *data center*, the proposed Fed-PAD framework introduces a *server* to exploit the fPAD information of all data centers by aggregating the above model updates $(\mathcal{W}^1, \mathcal{W}^2, ..., \mathcal{W}^K)$ of all data centers. Inspired by the Federated Averaging [14] algorithm, in the proposed framework, server carries out the aggregation of model updates via calculating the average of updated parameters $(\mathcal{W}^1, \mathcal{W}^2, ..., \mathcal{W}^K)$ in all data centers as follows:

$$\mathcal{W} = \frac{1}{K} \sum_{k=1}^{K} \mathcal{W}^k. \tag{2}$$

After the aggregation, server produces a global fPAD model parameterized by $\mathcal{W}$ that exploits the fPAD information of various data centers without accessing the private fPAD data. We can further extend the above aggregation process into t rounds. Server distributes the aggregated model $\mathcal{W}$ to every data center as the initial model parameters for the next-round updating of local parameters. Thus, data centers can obtain the t-th round updated parameters denoted as $(\mathcal{W}_t^1, \mathcal{W}_t^2, ..., \mathcal{W}_t^K)$. The t-th aggregation in the server can be carried out as follows:

$$\mathcal{W}_t = \frac{1}{K} \sum_{k=1}^{K} \mathcal{W}_t^k. \tag{3}$$

After t-rounds of communication between data centers and the server, the trained global fPAD model parameterized by $\mathcal{W}_t$ can be obtained without compromising the private data of individual *data centers*. Once training is converged, *users* will download the trained model from the server to their devices to carry out fPAD locally.

4. Experiments

To evaluate the performance of the proposed FedPAD framework, we carry out experiments using five 2D fPAD datasets and two 3D mask fPAD datasets. In this section, we first describe the datasets and the testing protocol used in our experiments. Then we report various experimental results based on multiple fPAD datasets. Discussions and analysis about the results are carried out to provide various insights about FL for fPAD.

4.1. Experimental Settings

4.1.1 Datasets

We evaluate our method using the following seven publicly available fPAD datasets which contain print, video replay and 3D mask attacks:

1) Oulu-NPU [2] (O for short)
2) CASIA-MFSD [24] (C for short)
3) Idiap Replay-Attack [4] (I for short)
4) MSU-MFSD [23] (M for short)
5) SiW [11] (S for short)
6) 3DMAD [5] (3 for short)
7) HKBUMARsV2 [9] (H for short).

Table 1 shows the variations in these seven datasets. Some sample images from these datasets are shown in Fig. 3.
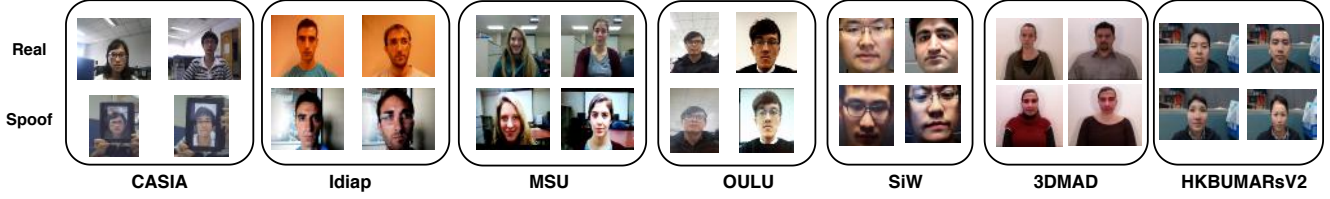
Figure 3. Sample images corresponding to real and attacked faces from CASIA-MFSD [24], Idiap Replay-Attack [4], MSU-MFSD [23], Oulu-NPU [2], SiW [11], 3DMAD [5], and HKBUMARsV2 [9] datasets.

Table 1. Comparison of seven experimental datasets.

Dataset	Extra light	Complex background	Attack type	Display devices
C	No	Yes	Printed photo Cut photo Replayed video	iPad
I	Yes	Yes	Printed photo Display photo Replayed video	iPhone 3GS iPad
M	No	Yes	Printed photo Replayed video	iPad Air iPhone 5S
O	Yes	No	Printed photo Display photo Replayed video	Dell 1905FP Macbook Retina
S	Yes	Yes	Printed photo Display photo Replayed video	Dell 1905FP iPad Pro iPhone 7 Galaxy S8 Asus MB168B
3	No	No	Thatsmyface 3D mask	Kinect
H	Yes	Yes	Thatsmyface 3D mask REAL-f mask	MV-U3B

From Table 1 and Fig. 3 it can be seen that different fPAD datasets exploit different characteristics and scenarios of face presentation attacks (*i.e.* different attack types, display materials and resolution, illumination, background and so on). Therefore, significant domain shifts exist among these datasets.

4.1.2 Protocol

The testing protocol used in the paper is designed to test the generalization ability of fPAD models. Therefore, in each test, performance of a trained model is evaluated against a dataset that it has not been observed during training. In particular, we choose one dataset at a time to emulate the role of *users* and consider all other datasets as *data centers*. Real images and spoof images of *data centers* are used to train a fPAD model. The trained model is tested considering the dataset that emulates the role of *users*. We evaluate the performance of the model by considering how well the model is able to differentiate between real and spoof images belonging to each *user*.

4.1.3 Evaluation Metrics

Half Total Error Rate (HTER) [1] (half of the summation of false acceptance rate and false rejection rate), Equal Er-

ror Rates (EER) and Area Under Curve (AUC) are used as evaluation metrics in our experiments, which are three most widely-used metrics for the cross-datasets/cross-domain evaluations. Following [12], in the absence of a development set, thresholds required for calculating evaluation metrics are determined based on the data in all *data centers*.

4.1.4 Implementation Details

Our deep network is implemented on the platform of PyTorch. We adopt Resnet-18 [6] as the structure of fPAD models $\mathcal{F}^i(i = 1, 2, 3, ..., K)$. The Adam optimizer [7] is used for the optimization. The learning rate is set as 1e-2. The batch size is 64 per data center. Local optimization epoch L is set equal to 3.

4.2. Experimental Results

In this section we demonstrate the practicality and generalization ability of the proposed framework in the real-world scenario. We first compare the performance of the proposed framework with models trained with data from a single data center. As mentioned above, due to the limitation of data privacy that exists in the real-world, data cannot be shared between different *data centers*. In this case, *users* will directly obtain a trained model from one of the *data centers*. We report the performance of this baseline in the Table 2 under the label **Single**. For different choices of user datasets (from O, C, I, M), we report the performance when the model is trained from the remaining datasets independently.

Rather than obtaining a trained model from a single data center, it is possible for users to obtain multiple trained models from several data centers and fuse their prediction scores during inference, which is also privacy preserving. In this case, we fuse the prediction scores of the trained model from various data centers by calculating the average. The results of this baseline are shown in Table 2 denoted as **Fused**. Please note that it is impossible for users to carry out feature fusion because a classifier cannot be trained based on the fused features without accessing to any real/spoof data during inference in the users. According to Table 2, fusing scores obtained from different data centers improves

Table 2. Comparison with models trained by data from single data center and various data centers.

| Methods	Data Centers	User	HTER (%)	EER (%)	AUC (%)	Avg. HTER	Avg. EER	Avg. AUC
Single	O	M	41.29	37.42	67.93	36.43	34.31	70.36
	C	M	27.09	24.69	82.91			
	I	M	49.05	20.04	85.89			
	O	C	31.33	34.73	73.19			
	M	C	39.80	40.67	66.58			
	I	C	49.25	47.11	55.41			
	O	I	42.21	43.05	54.16			
	C	I	45.99	48.55	51.24			
	M	I	48.50	33.70	66.29			
	M	O	29.80	24.12	84.86			
	C	O	33.97	21.24	84.33			
	I	O	46.95	35.16	71.58			
Fused	O&C&I	M	34.42	23.26	81.67	35.75	31.29	73.89
	O&M&I	C	38.32	38.31	67.93			
	O&C&M	I	42.21	41.36	59.72			
	I&C&M	O	28.04	22.24	86.24			
Ours	O&C&I	M	19.45	17.43	90.24	**32.17**	**28.84**	**76.51**
	O&M&I	C	42.27	36.95	70.49			
	O&C&M	I	32.53	26.54	73.58			
	I&C&M	O	34.44	34.45	71.74			
All (Upper Bound)	O&C&I	M	21.80	17.18	90.96	27.26	25.09	80.42
	O&M&I	C	29.46	31.54	76.29			
	O&C&M	I	30.57	25.71	72.21			
	I&C&M	O	27.22	25.91	82.21			
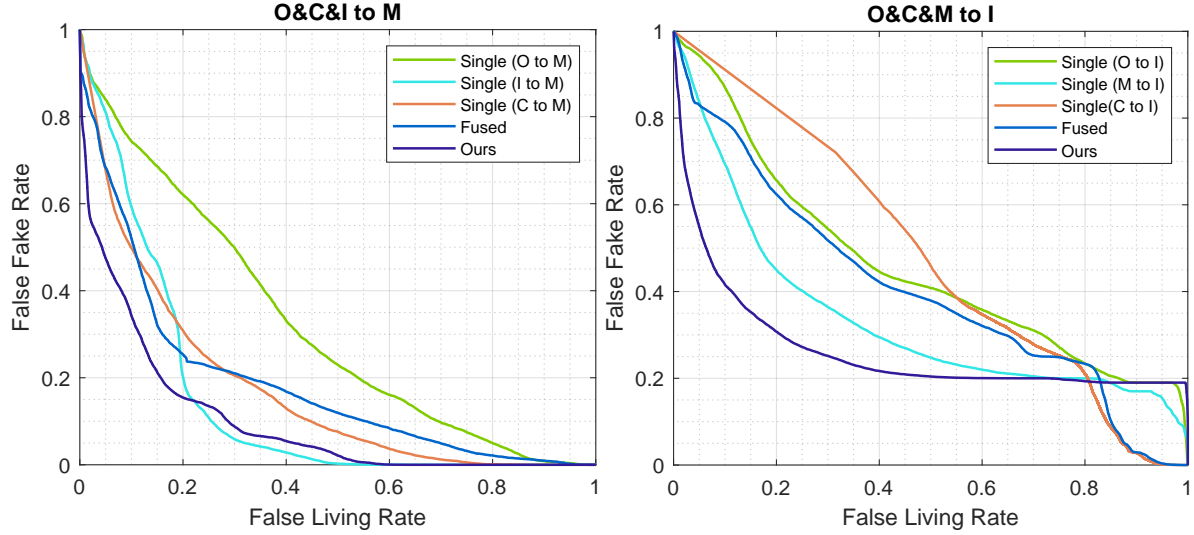
Figure 4. ROC curves of models trained by data from single data center and various data centers.

the fPAD performance on average. However, this would require higher inference time and computation complexity (of order three for the case considered in this experiment).

On the other hand, **Ours** shows the results obtained by the proposed FedPAD framework. Table 2 illustrates that the average values of all evaluation metrics of the proposed

framework outperform both baselines. Specifically, we further plot the ROC curves which choose MSU-MFSD (M) and Idiap Replay-Attack (I) datasets as the users as two examples, respectively in Fig 4. Fig 4 also shows that the proposed FedPAD framework performs better than two baselines. This demonstrates that the proposed method is
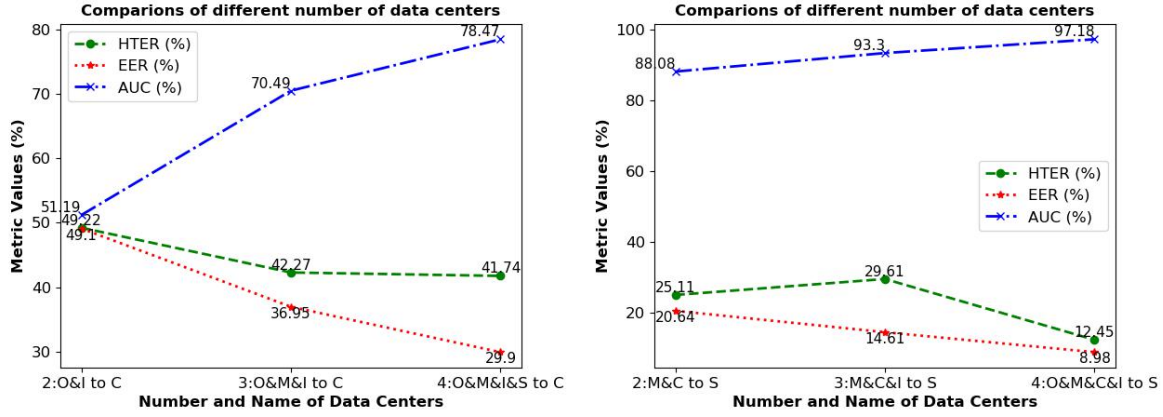
Figure 5. Comparison of different number of data centers.

Table 3. Effect of using different types of spoof attacks

Methods	Data Centers	User	HTER (%)	EER (%)	AUC (%)
Single	I (Print)	M (Print, Video)	38.82	33.63	72.46
	O (Video)	M (Print, Video)	35.76	28.55	78.86
Fused	I (Print) & O (video)	M (Print, Video)	35.22	**25.56**	81.54
Ours	I (Print) & O (video)	M (Print, Video)	**30.51**	26.10	**84.82**

more effective in exploiting fPAD information from multiple data centers. This is because the proposed framework actively combines fPAD information across data centers during training as opposed to the **fused** baseline. As a result, it is able to generalize better to unseen/novel spoof attacks.

Moreover, we further consider the case where a model is trained with data from all available data centers, which is denoted as **All** in Table 2. Note that this baseline violates the assumption of preserving data privacy, and therefore is not a valid comparison for FedPAD. Nevertheless, it indicates the upper bound of performance for the proposed FedPAD framework. From Table 2, it can be seen that the proposed FedPAD framework is only 3.9% worse than the upper bound in terms of AUC. This shows the proposed framework is able to obtain a privacy perseserving fPAD model without sacrificing too much fPAD performance. This result verifies the practicality of the proposed framework.

Figure 6. ROC curves of models trained by different types of 2D spoof attacks.

4.2.1 Comparison of different number of data centers

In this section, we investigate the importance of having more data centers during training. Different data centers exploit different characteristics of face presentation attacks. Therefore, we expect aggregating information from more data centers in the proposed FedPAD framework to produce more robust models with better generalization. In order to verify this point, we increase the number of data centers in
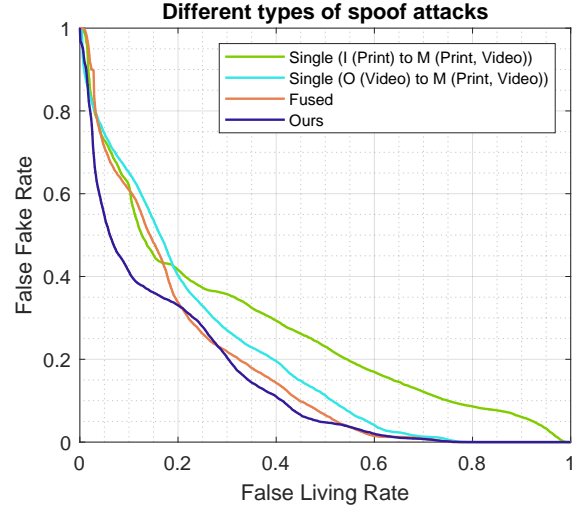
the proposed FedPAD framework and report the results in Fig. 5. The experiments are carried out using five datasets (O, M, I, C, S). In Fig. 5 (left), we select the dataset C as the data presented to the user and the remaining datasets as the data centers for training the fPAD model with our FedPAD framework. We increase the number of data centers from 2 to 4 and corresponding *data centers* are shown in the X-axis. Another experiment is carried out with a different combination of the same five datasets and the results are

Table 4. Impact of adding data centers with diverse attacks

Data Centers	User	HTER (%)	EER (%)	AUC (%)
O&C&I&M (2D)	H (3D)	47.02	18.31	85.06
O&C&I&M (2D)&3 (3D)	H (3D)	**34.70**	**14.20**	**92.35**

shown in Fig. 5 (right). From the curve in Fig. 5, it can be seen that most values of evaluation metrics improve along when the number of data centers increases. This demonstrates that increasing the number of data centers in the proposed FedPAD framework can improve the performance.

4.2.2 Generalization ability to various 2D spoof attacks

In reality, due to limited resources, one data center may be only able to collect limited types of 2D attacks. However, various 2D attacks may appear to the users. This section supposes that one data center collects one particular type of 2D attack such as print attack or video-replay attack. As illustrated in Table 3, first, we select real faces and print attacks from dataset I and real faces and video-replay attacks from dataset O to train a fPAD model respectively and evaluate them on dataset M (containing both print attacks and video-replay attacks). In both considered cases as shown in Table 3, the corresponding trained models cannot generalize well to dataset M which contains the additional types of 2D attacks compared to dataset I and O, respectively. This tendency can be alleviated when the prediction scores of two independently trained models on both types of attacks are fused as show in Table 3. Comparatively, FedPAD method obtains a performance gain of 4.71% in HTER and 3.3% in AUC compared to score fusion. We also plot the corresponding ROC curve for this comparison in Fig 6, which also demonstrate the superior performance of the proposed method. This experiment demonstrates that carrying out FedPAD framework among data centers with different types of 2D spoof attacks can improve the generalization ability of the trained fPAD model to various 2D spoof attacks.

4.2.3 Generalization ability to 3D mask attacks

In this section, we investigate the generalization ability of the proposed FedPAD framework to 3D mask attacks. First, a fPAD model is trained with data centers exploiting 2D attacks (data from datasets O, C, I and M). This model is tested with 3D mask attacks (data from dataset H). Then, we include one more data center containing 3D mask attacks (dataset 3) into our FedPAD framework and retrain our model. Table 4 and Fig 7 show that introducing diversity of data centers (by including a 3D mask attack) can significantly improve performance in all evaluation metrics. This experiment demonstrates that increasing data centers with 3D mask attacks within the proposed FedPAD frame-
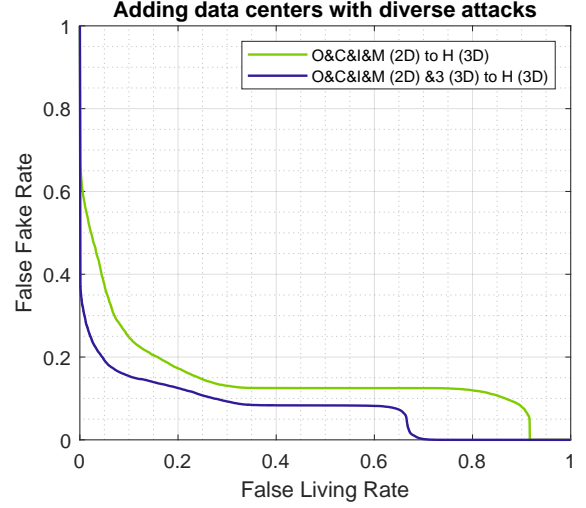

Figure 7. ROC curves of models trained with added data center of 3D mask attacks.

work can improve the generalization ability of the trained model to the novel 3D mask attacks.

5. Conclusion

In this paper, we presented FedPAD, a federated learning-based framework targeting application of fPAD with the objective of obtaining generalized fPAD models while preserving data privacy. Through communications between *data centers* and the *server*, a globe fPAD model is obtained by iteratively aggregating the model updates from various *data centers*. Local private data is not accessed during this process. Extensive experiments are carried out to demonstrate the effectiveness of the proposed framework which provide various insights regarding federated learning for fPAD.

In our experiments, we encountered situations where adding more data centers slightly decreased the performance (HTER value of Fig. 5 (right) when the data center increases from 2 to 3). Adding more data centers into our framework not only increases the diversity of fPAD information but also intensifies domain shift among data centers. This domain shift may increase the difficulties in exploiting an optimal fPAD model in the federated learning process. In the future, we will explore an improved federated learning method that tackles data centers with significant domain shift effectively.

References

[1] Samy Bengio and Johnny Mariéthoz. A statistical significance test for person authentication. In *The Speaker and Language Recognition Workshop*, 2004.

[2] Zinelabinde Boulkenafet and et al. Oulu-npu: A mobile face presentation attack database with real-world variations. In *FG*, 2017.

[3] Zinelabidine Boulkenafet, Jukka Komulainen, and Abdenour Hadid. Face spoofing detection using colour texture analysis. In *IEEE Trans. Inf. Forens. Security, 11(8): 1818-1830*, 2016.

[4] Ivana Chingovska, André Anjos, and Sébastien Marcel. On the effectiveness of local binary patterns in face anti-spoofing. In *BIOSIG*, 2012.

[5] N. Erdogmus and S. Marcel. Spoofing face recognition with 3D masks. 2014. TIFS.

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

[7] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *arXiv preprint arXiv:1412.6980*, 2014.

[8] Tian Li, Anit Kumar Sahu, Ameet Talwalkar, and Virginia Smith. Federated learning: Challenges, methods, and future directions. In *arXiv preprint arXiv:1908.07873*, 2019.

[9] Siqi Liu, Xiangyuan Lan, and Pong C. Yuen. Remote photoplethysmography correspondence feature for 3D mask face presentation attack detection. In *ECCV*, 2018.

[10] Siqi Liu, Pong C. Yuen, Shengping Zhang, and Guoying Zhao. 3D mask face anti-spoofing with remote photoplethysmography. In *ECCV*, 2016.

[11] Yaojie Liu, Amin Jourabloo, and Xiaoming Liu. Learning deep models for face anti-spoofing: Binary or auxiliary supervision. In *CVPR*, 2018.

[12] Yaojie Liu, Joel Stehouwer, Amin Jourabloo, and Xiaoming Liu. Deep tree learning for zero-shot face anti-spoofing. In *CVPR*, 2019.

[13] Jukka Määttä, Abdenour Hadid, and Matti Pietikäinen. Face spoofing detection from single images using micro-texture analysis. In *IJCB*, 2011.

[14] H Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, et al. Communication-efficient learning of deep networks from decentralized data. In *AISTATS*, 2016.

[15] Mehryar Mohri, Gary Sivek, and Ananda Theertha Suresh. Agnostic federated learning. 2019.

[16] Tiago Freitas Pereira and et al. Face liveness detection using dynamic texture. In *EURASIP Journal on Image and Video Processing, (1): 1-15*, 2014.

[17] Anit Kumar Sahu, Tian Li, Maziar Sanjabi, Manzil Zaheer, Ameet Talwalkar, and Virginia Smith. On the convergence of federated optimization in heterogeneous networks. In *arXiv preprint arXiv:1812.06127*, 2018.

[18] Rui Shao, Xiangyuan Lan, Jiawei Li, and Pong C. Yuen. Multi-adversarial discriminative deep domain generalization for face presentation attack detection. In *CVPR*, 2019.

[19] Rui Shao, Xiangyuan Lan, and Pong C. Yuen. Deep convolutional dynamic texture learning with adaptive channel-discriminability for 3D mask face anti-spoofing. In *IJCB*, 2017.

[20] Rui Shao, Xiangyuan Lan, and Pong C. Yuen. Joint discriminative learning of deep dynamic textures for 3D mask face anti-spoofing. In *IEEE Trans. Inf. Forens. Security, 14(4): 923-938*, 2019.

[21] Rui Shao, Xiangyuan Lan, and Pong C. Yuen. Regularized fine-grained meta face anti-spoofing. In *AAAI*, 2020.

[22] Virginia Smith, Chao-Kai Chiang, Maziar Sanjabi, and Ameet S Talwalkar. Federated multi-task learning. In *NIPS*, 2017.

[23] Di Wen, Hu Han, and Anil K. Jain. Face spoof detection with image distortion analysis. In *IEEE Trans. Inf. Forens. Security, 10(4): 746-761*, 2015.

[24] Zhiwei Zhang and et al. A face antispoofing database with diverse attacks. In *ICB*, 2012.