Progressive Transfer Learning for Face Anti-Spoofing

Ruijie Quan, Yu Wu, Xin Yu, and Yi Yang, *Senior Member, IEEE*

***Abstract*—Face anti-spoofing (FAS) techniques play an important role in defending face recognition systems against spoofing attacks. Existing FAS methods often require a large number of annotated spoofing face data to train effective anti-spoofing models. Considering the attacking nature of spoofing data and its diverse variants, obtaining all the spoofing types in advance is difficult. This would limit the performance of FAS networks in practice. Thus, an online learning FAS method is highly desirable.**

In this paper, we present a semi-supervised learning based framework to tackle face spoofing attacks with only a few labeled training data (*e.g.*, $\sim$ 50 face images). Specifically, we progressively adopt the unlabeled data with reliable pseudo labels during training to enrich the variety of training data. We observed that face spoofing data are naturally presented in the format of video streams. Thus, we exploit the temporal consistency to consolidate the reliability of a pseudo label for a selected image. Furthermore, we propose an adaptive transfer mechanism to ameliorate the influence of unseen spoofing data. Benefiting from the progressively-labeling nature of our method, we are able to train our network on not only data of seen spoofing types (*i.e.*, the source domain) but also unlabeled data of unseen attacking types (*i.e.*, the target domain). In this way, our method can reduce the domain gap and is more practical in real-world anti-spoofing scenarios. Extensive experiments in both the intra-database and inter-database scenarios demonstrate that our method is on par with the state-of-the-art methods but employs remarkably less labeled data (less than 0.1% labeled spoofing data in a dataset). Moreover, our method significantly outperforms fully-supervised methods on cross-domain testing scenarios with the help of our progressive learning fashion.

***Index Terms*—Face Anti Spoofing, Progressive Learning, Transfer Learning.**

I. Introduction

FACE recognition systems [1], [2] have been widely deployed in many real-world scenarios. For example, access authorization systems usually verify identity information via face recognition/verification. Deep convolutional neural networks (CNNs) have attained promising success in recognizing different faces. However, these face models are often vulnerable to face spoofing attacks, *i.e.*, printed faces (print attack) or replayed faces on a digital device (replay attack). Attackers utilize such attacks to fool existing face recognition systems, leading to severe privacy breaching and financial damages. As a result, face anti-spoofing (FAS) methods have been exploited as a prerequisite procedure in many face recognition/verification applications.

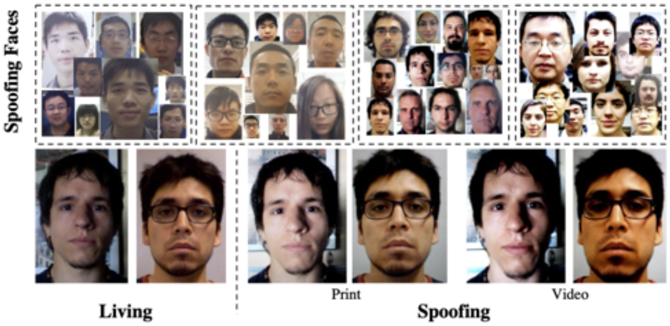

Fig. 1. Spoofing faces emerge constantly and are difficult to distinguish. Considering various illumination conditions and diverse capturing systems, it would be even more challenging to identify whether a picture captures a living face or a spoofing one. Therefore, labeling spoofing data is also difficult.

Recent FAS methods mainly focus on the exploration of supervised approaches [3], [4], [5], [6]. Thus, existing methods require a large number of spoofing data annotations in order to train sufficiently discriminative networks. However, it is difficult for human annotators to distinguish whether an image captures a living face or a spoofing one due to the similarity between living faces and spoofing ones, as shown in Figure. 1.

In addition, as attacks to face recognition systems also evolve, new forms of spoofing faces are likely created and severely degrade the performance of FAS systems. Hence, fully supervised FAS systems require continuous data labeling to tackle new spoofing attacks. Doing so not only is laborious and costly but also cannot handle new attacks promptly.

In this paper, we present a semi-supervised learning based framework, namely progressive transfer learning, to address face spoofing attacks with only a few labeled training data. Different from existing methods, our proposed method gradually selects unlabeled data with highly-confident pseudo labels to enrich the variety of training data. To be specific, we firstly optimize our model by using the small number of labeled spoofing data and then select highly-confident data from the rest of unlabeled ones to update our model in a progressive fashion. In doing so, our method firstly adopts easy and reliable pseudo-labeled spoofing data for model update and then explores difficult ones as the discriminativeness ability of our model improves.

To obtain reliable pseudo labeled data for training, we design a new strategy in our data selection. Since spoofing face data are naturally presented in the format of video streams, we exploit the temporal consistency of video labels as a constraint

This work was supported by the Australian Research Council's Discovery Projects funding scheme under Project DP200100938. Yu Wu is supported by the Google PhD Fellowship. *(Corresponding Author: Yu Wu.)*

Ruijie Quan, Yu Wu, Xin Yu and Yi Yang are with the ReLER Lab, Australian Artificial Intelligence Institute, University of Technology Sydney, Ultimo, NSW 2007, Australia (e-mail: ruijie.quan@student.uts.edu.au; yu.wu-3@student.uts.edu.au; xin.yu@uts.edu.au; yi.yang@uts.edu.au).

in evaluating the reliability of sampled images. For instance, as seen in Figure. 2, due to the drastic illumination changes, our model might predict different pseudo labels for different frames. This confidence inconsistency within a video segment gives us a clue that a video might not be a reliable one. Motivated by this observation, we evaluate the pseudo label confidence of a frame in a video segment by averaging the mean confidence score of all the other frames in the video and the confidence score of the chosen frame. Then, we choose frames with high average confidence score (*i.e.*, consistent pseudo labels across an entire video). Using our temporal confidence consistency mechanism, we significantly improve the robustness and reliability of selected pseudo labels.

Considering the spoofing nature, some attacking types might be totally new to a trained FAS network and will degrade its performance dramatically. Benefiting from our progressive learning manner, our method can exploit unlabeled data of unseen attacking types to reduce the domain gap. To be specific, our method at first trains a network with source domain data and then gradually incorporates unlabeled target domain data. Note that, living and spoofing faces in the target domain are mixed together (resembling the data stream in real-world applications). In order to ameliorate the domain bias and stabilize the anti-spoofing performance, we further propose an adaptive transfer mechanism on unlabeled data from the target domain. Specifically, we design a dynamic weight to increase the contribution of unlabeled target domain data in accordance with the iteration steps while preserving the impact of source domain data. With the help of our adaptive transfer mechanism, our method is able to handle unseen attacking types in an online and unsupervised manner.

Extensive experiments demonstrate that our proposed method is effective in both intra-database and inter-database testing scenarios. For the intra-database scenario, the train and test sets belong to the same dataset, and they contain same attacking types as well as similar data distributions. Therefore, our framework aims to address face spoofing attacks with only a few human-annotated data. Whereas training data (source domain) and testing data (target domain) are from different datasets in the inter-database scenario, they have obvious domain gap, *e.g.*, unseen attacking types, different illumination conditions, background scenes and recording devices, some examples are shown in Figure. 1. In the intra-database scenario, we only require around 50 human-annotated (<1‰) spoofing face data, and achieve competitive performance compared to state-of-the-art fully-supervised methods. Furthermore, our method obtains state-of-the-art performance in the inter-database testing without leveraging any labels from the target domain data. This also makes our method more appealing in real-world applications since our model can learn from unlabeled spoofing data in unseen attacking types.

Overall, our contributions are summarized as follows:

- We propose a semi-supervised learning based framework to tackle face spoofing attacks with only a few labeled data instead of relying on tedious data annotations.
- We exploit a temporal consistency constraint to verify the reliability of pseudo labels of selected data, thus significantly facilitating our network training.
- We design an adaptive transfer mechanism to ameliorate the domain bias by gradually increasing the contribution of unlabeled target domain data in training.
- Our method only uses a few labeled data yet achieves state-of-the-art performance on both the intra-database testing and the inter-database testing scenarios.

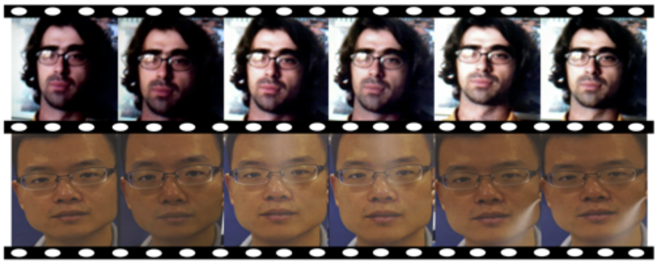
Fig. 2. Face spoofing data are often presented in the format of video streams. It might be hard to determine whether the first two frames in each row are living ones or spoofing ones. The temporal inconsistency implies that videos might be spoofing ones, since living videos in general do not suffer from abrupt illumination changes.

II. Related Work

A. Face Anti-Spoofing

Face anti-spoofing has been considered indispensable in face recognition systems. Previous face anti-spoofing methods are mainly grouped into two categories. Some early approaches detect specific facial motion patterns (*i.e.*, eye blinking, mouth movements, and facial expression changes) as the evidence of face liveliness [7], [8], [9], [10]. However, these methods might fail when encountering novel attack types such as video replay. Moreover, these methods need relatively long time to detect all those moving actions. Another category of prior works extract hand-crafted features from captured images and train a binary classifier to discriminate between spoofing data and living ones [11], [12], [13], [14], [15], [16].

Recently, deep learning based methods have demonstrated their superiority in determining spoofing faces. Zhang *et al.* [3] proposed a multi-modal fusion method by leveraging multiple visual modalities on their presented dataset. Yang *et al.* [4] presented a spatio-temporal anti-spoofing network by considering both global temporal and local spatial information to distinguish living faces from spoofing ones. Liu *et al.* [17] created a large dataset with 13 different types of spoofing attacks and used a deep tree network to recognize spoofing attacking types. These deep learning-based methods generally perform well on seen face spoofing data but suffer performance degradation when new attacking types appear.

Although recent deep learning FAS approaches have demonstrated superior performance, they heavily rely on extensive human-labeled data. Furthermore, those fully supervised methods might suffer severe performance degradation when new forms or styles [18] of spoofing faces emerge. In contrast, our method tackles face spoofing attacks with only a few labeled training data. In addition, it progressively exploits unlabeled data of unseen attacking types, and thus reduces the domain gap between the source domain and the target domain data.

B. Progressive Paradigm

Bengio *et al.* [19] proposed an easy-to-hard learning strategy to train machine learning models, called curriculum learning. Then Kumar *et al.* [20] proposed a self-paced learning algorithm based on curriculum learning to select samples in an iterative framework. Khan *et al.* [21] proved that curriculum learning and the human learning principles are consistent. Recently, curriculum learning has been applied to semi-supervised image classification [22], question answering [23], [24], person reidentification [25], [26], action recognition [27], [28], etc.

III. Methodology

Inspired by the existing self-paced learning and curriculum learning algorithms, we propose a progressive transfer learning method for FAS. In this section, we first introduce the preliminaries of our progressive transfer learning for FAS in Sec. III-A. Then, we present our progressive learning framework equipped with the designed temporal constraint mechanism in the intra-database scenario in Sec. III-B. Moreover, we extend our progressive transfer learning to the inter-database scenario. An adaptive transfer mechanism is incorporated in the inter-database framework, and we introduce it in Sec. III-C.

A. Preliminaries

Existing FAS methods [5], [17] require a large amount of human-annotated training data $\{(x_i, y_i)\}_{i=1}^{N}$, where x_i denotes the i-th of N samples associated with the label y_i. The label y_i indicates whether x_i is a living face or a spoofing one. A CNN model with a binary classifier is trained to detect spoofing faces and outputs prediction scores $\langle \alpha_i, \beta_i \rangle$ for each face data x_i, where α_i and β_i indicate the prediction confidence scores of a living face and a spoofing one, respectively. Different from previous methods that entirely rely on the annotated data, our method only utilizes S labeled samples denoted as the labeled subset $\mathcal{L}$. Then we jointly train the model $\Phi(\cdot)$ on the labeled data $\mathcal{L}$ and the tremendous unlabeled data $\mathcal{U}$ with the help of the progressive learning framework.

$$\mathcal{L} = \{(x_i, y_i)\}_{i=1}^{S}, \qquad \mathcal{U} = \{x_i\}_{i=S+1}^{N},$$

where $S \ll N$. We will discuss and evaluate different values of S in the experiments section. During the label estimating stage, we assign a pseudo label $\hat{y}_i$ for each $x_i \in \mathcal{U}$ according to the prediction confidence scores. Let $\mathcal{A}_t$ denote the selected pseudo-labeled data at the t-th iteration,

$$\mathcal{A}_t = \{(x_j, \hat{y}_j) \mid \delta(j) = 1\}_{j}^{C_t}, \tag{1}$$

where C_t is the sampling size at t-th iteration, $\hat{y}_j$ is the pseudo label for the face image x_j, and δ is an indication function to indicate whether x_j belongs to the selected data or not. $\delta(j) = 1$ indicates that the data x_j is a selected sample, and $\delta(j) = 0$ means that the data x_j has not been chosen.

B. Intra-database Anti-spoofing Scenario

To address face spoofing attacks with only a few labeled training data, we present a semi-supervised learning based framework, named progressive transfer learning. We first introduce it in the intra-database scenario, where the training and test sets have similar data distributions.

Algorithm 1 Intra-database Progressive Learning Algorithm

```
Input: The selected labeled training data $\mathcal{L}$ of $S$ samples,
    remaining unlabeled training data $\mathcal{U}$ of $N$-$S$ samples, testing
    data $\mathcal{T}$, initialized CNN model $\Phi_0(\cdot)$.
Output: Final CNN model $\Phi_T(\cdot)$.
  Update the CNN model $\Phi_1(\cdot) \leftarrow \Phi_0(\mathcal{L})$
  Estimate the pseudo label $\hat{y_i}$ for $\mathcal{U}$ and then select the
  pseudo-labeled data $\mathcal{A}_0$ by setting the sampling size $C_t \leftarrow$
  $\sigma_t \cdot N$, and iteration $t \leftarrow 0$
  while $C_t \leq N$ do
    $t \leftarrow t + 1$. Update training set: $\mathcal{D}_t \leftarrow \mathcal{L} \cup \mathcal{A}_{t-1}$
    Update the CNN model $\Phi_t(\cdot) \leftarrow \Phi_{t-1}(\mathcal{D}_t)$
    Update the sampling size: $C_t \leftarrow \sigma_t \cdot N \cdot t$
    Evaluate the model $\Phi_t(\cdot)$ on $\mathcal{U}$ and obtain prediction
      results $\langle \alpha_i, \beta_i \rangle$ for each unlabeled face $x_i \in \mathcal{U}$
    Generate $\mathcal{A}_t$ by selecting top-$C_t$ samples, $|\mathcal{A}_t| \leftarrow C_t$,
      when the final prediction confidence score $p'_i > \mu$
  end while
  Evaluate $\Phi_T(\cdot)$ on the testing data $\mathcal{T}$
```

Since only a few labeled data are available to our method, our CNN model is not discriminative enough to distinguish difficult spoofing face data at the beginning. Thus, our framework works in a progressive fashion by iteratively converting the unlabeled data into the pseudo-labeled ones from easy to hard. Specifically, we first utilize a few ($S \ll N$) labeled faces to optimize the initial CNN model $\Phi_0(\cdot)$. Then the model is updated iteratively by: (i) estimating pseudo labels for the unlabeled data and selecting highly-confident ones to constitute a pseudo-labeled subset $\mathcal{A}$ according to the prediction confidence scores; (ii) updating the model with both the labeled faces $\mathcal{L}$ and the reliably pseudo-labeled faces $\mathcal{A}$. The new training set $\mathcal{D}_t$ is updated by $\mathcal{D}_t \leftarrow \mathcal{L} \cup \mathcal{A}$ at the t-th iteration. Finally, after obtaining the model jointly trained on the labeled data $\mathcal{L}$ and the pseudo-labeled data $\mathcal{A}$, we evaluate our model on the testing data $\mathcal{T}$.

At each iteration of our progressive learning framework, we incorporate unlabeled data with highly-confident pseudo labels into the training data to enrich its variety. To be specific, after the CNN model predicts confidence scores on the unlabeled data $\mathcal{U}$ at t-th iteration, we select C_t samples from $\mathcal{U}$ as the highly-confident pseudo-labeled data $\mathcal{A}_t$, where $C_t = \sigma_t \cdot N$. The C_t samples are selected from $\mathcal{U}$ according to their prediction confidence scores in a descending order. The coefficient σ_t is the sample rate associated with the iteration number t. We define $\sigma_t = k \cdot t$ in our progressive learning framework, and k is set to 0.08 in our experiments. The impact of different k values are investigated in the experiment. For each selected face $x_i \in \mathcal{A}_t$, we estimate its pseudo label $\hat{y}_i$ and its confidence score p_i as:

$$p_i = max(\alpha_i, \beta_i), \tag{2}$$

$$\hat{y}_i \leftarrow \arg\max(\alpha_i, \beta_i). \tag{3}$$

Additionally, we filter out low confident data whose prediction confidence score p'_i is less than a threshold μ during the data selection procedure. The threshold is utilized to ensure the reliability of the pseudo labels of the selected data. We also study the influences of different values of μ in experiments.

As shown in Alg. 1, we exploit the unlabeled data $\mathcal{U}$ by transforming $\mathcal{U}$ to $\mathcal{A}$ progressively. To further guarantee the stability of our progressive learning algorithm, we require that both living data and spoofing data in the selected data are larger than $\frac{C_t}{100}$. If the requirement is not met, our progressive learning algorithm will turn into the next iteration to select more data without updating our model. Once the requirement is met, we will update our model. Such a strategy can prevent that the selected data all are either living faces or spoofing ones. Otherwise, the classifier of the model may fail to learn a good classification boundary from the extremely unbalanced data distribution, and thus the model will fail to converge.

Temporal Constraint. We observe that spoofing face data are naturally presented in the format of video streams and faces in a video segment have a same spoofing label. However, due to illumination changes, the predicted confidence scores of frames in the same video segment might vary widely. As we assign the pseudo label for the unlabeled data according to the prediction confidence score, unreliable confidence scores will affect the reliability of pseudo labels. To tackle the problem, we further propose a temporal constraint mechanism to leverage the temporal consistency as a constraint. Therefore, the confidence inconsistency within a video segment can give us a clue that the video segment might not be a reliable one.

We assume there are M face images in a video segment, and thus the labels of the M faces should be the same (either all living or all spoofing). Let p_m be the prediction confidence score of the face image x_m at the m-th frame of the video segment and we obtain the averaged prediction confidence score $\bar{p}$ by: $\bar{p} = \frac{\sum(p_1, p_2, p_3, \ldots, p_M)}{M}$. The averaged prediction confidence score can be regarded as a temporal context consistency information. A simple strategy is to take the averaged prediction confidence score $\bar{p}$ as the new confidence score of all frames. However, since we select samples according to the prediction confidence scores and faces of a same video segment keep the same confidence score $\bar{p}$, the model would select many similar faces from the same video segment during the unlabeled data selection procedure. Therefore, the diversity of the selected data will significantly degrade, thus leading to a poor classification boundary especially at first several iterations. We then propose to combine the prediction confidence score of the frame itself and the average confidence of all the other related frames as the final prediction confidence score p'_i for data x_i, defined as: $p'_i = \frac{(p_i + \bar{p})}{2}$.

With the help of our designed temporal constraint mechanism, our method is able to select reliable and diverse face data in the progressive learning. Thus, we alleviate prediction errors and improve pseudo-label robustness.

C. Inter-database Anti-spoofing Scenario

New attacking types of spoofing data emerge constantly and they might be totally new to a trained FAS network, leading to performance degradation significantly. Our progressive transfer learning network can be easily extended to tackle the problem in an online and unsupervised manner. Specifically, to reduce the domain gap between the seen data (source domain) and the newly emerged data (target domain), our method at first trains a network with source domain data and then gradually incorporates unlabeled target domain data via our progressive transfer learning approach.

```
Algorithm 2 Inter-database Progressive Learning Algorithm
Input: Labeled source domain data $\mathcal{L}$, remaining unlabeled
    source domain data $\mathcal{U}$, target domain training data $\mathcal{T}_{train}$ ($N$
    samples), target domain testing data $\mathcal{T}_{test}$, model $\Phi^0_{intra}(\cdot)$.
Output: The final CNN model $\Phi^{T'}_{inter}(\cdot)$.
  Update the CNN model $\Phi^1_{intra}(\cdot) \leftarrow \Phi^0_{intra}(\mathcal{L})$
  Implement Algorithm 1 on the source domain data to
  assign pseudo-labels to the unlabeled source domain data
  ($\mathcal{U} \rightarrow \hat{\mathcal{U}}$) and then initialize the CNN model for the inter-
  database scenario $\Phi^0_{inter}(\cdot) \leftarrow \Phi^T_{intra}(\cdot)$
  Estimate the pseudo label $\hat{y}_i$ for $\mathcal{T}_{train}$ and then select
  the pseudo-labeled data $\mathcal{A}_0$ by setting the sampling size
  $C_{t'} \leftarrow \sigma_{t'} \cdot N$, and iteration step $t' \leftarrow 0$
  while $C_{t'} \leq N$ do
    $t' \leftarrow t' + 1$
    Update training set: $\mathcal{D}_{t'} \leftarrow (\mathcal{L} \cup \hat{\mathcal{U}}) \cup \mathcal{A}_{t'-1}$
    Update the CNN model $\Phi^{t'}_{inter}(\cdot) \leftarrow \Phi^{t'-1}_{inter}(\mathcal{D}_{t'})$
    Update the sampling size: $C_{t'} \leftarrow \sigma_{t'} \cdot N \cdot t'$
    Evaluate $\Phi^{t'}_{inter}(\cdot)$ on $\mathcal{T}_{train}$ and obtain prediction result
      $\langle \alpha_i, \beta_i \rangle$ for each unlabeled face $x_i \in \mathcal{T}_{train}$
    Generate $\mathcal{A}_{t'}$ by selecting top-$C_{t'}$ samples, $|\mathcal{A}_{t'}| \leftarrow C_{t'}$,
      when the final prediction confidence score $p'_i > \mu$
  end while
  Evaluate $\Phi^{T'}_{inter}(\cdot)$ on the testing data $\mathcal{T}_{test}$
```

As illustrated in Alg. 2 and Figure. 3, our method firstly trains a model $\Phi_{intra}(\cdot)$ on $\mathcal{L}$ and $\mathcal{U}$ via our progressive learning algorithm of the intra-database scenario. After all the unlabeled source domain data $\mathcal{U}$ are pseudo-labeled ($\mathcal{U} \rightarrow \hat{\mathcal{U}}$) and are incorporated into the training data, we obtain our final intra-database model $\Phi^T_{intra}(\cdot)$ updated for T iterations. Then we leverage the model $\Phi^T_{intra}(\cdot)$ to initialize the model Φ^0_{inter} of the inter-database scenario, denoted as $\Phi^0_{inter}(\cdot) \leftarrow \Phi^T_{intra}(\cdot)$. The model updates iteratively by (1) selecting the highly-confident unlabeled target domain data $\mathcal{A}_{t'}$ with pseudo labels at the t'-th iteration, (2) jointly training on the source domain data $\mathcal{S}$ ($\mathcal{S} = \mathcal{L} \cup \hat{\mathcal{U}}$) and the pseudo-labeled target domain subset $\mathcal{A}_{t'}$. Note that, only a few data of the source domain are labeled in the intra-database stage. Finally, we evaluate $\Phi^{T'}_{inter}(\cdot)$ on the test data of target domain $\mathcal{T}_{test}$ after updating the model on the source domain data $\mathcal{S}$ and the pseudo-labeled data $\mathcal{A}_{T'}$ for total T' iterations.

In the inter-database scenario, we also use a threshold μ to determine the reliability of pseudo labels of the selected data. Furthermore, we adopt the same strategy introduced in the intra-database scenario to guarantee a minimum number (*i.e.*, $\frac{C_t}{100}$) of the living and spoofing data in each iteration. Moreover, the designed temporal constraint mechanism is also exploited to improve the reliability.

Adaptive Transfer. A trained model on the source domain

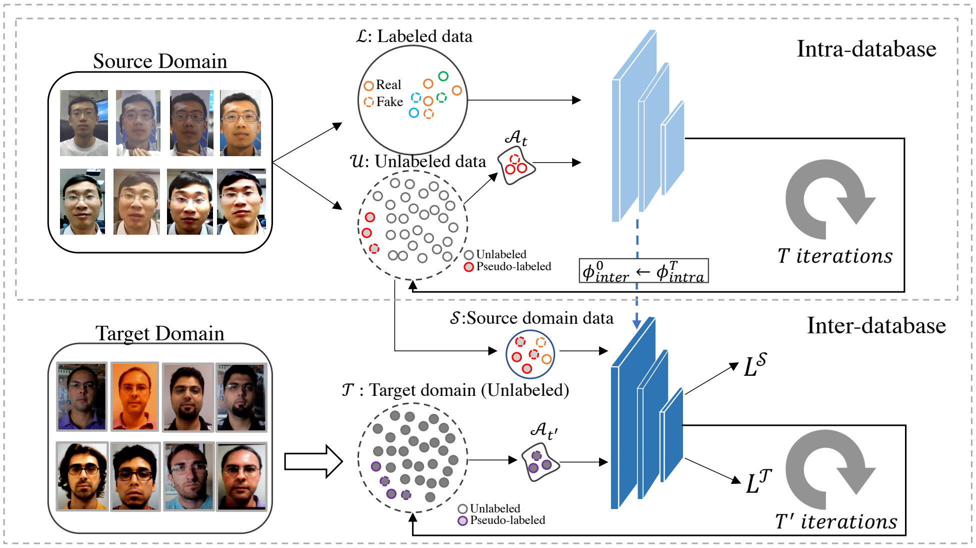

Fig. 3. Overview of our proposed progressive transfer learning framework in the inter-database scenario. It leverages two stages to update the CNN model: the intra-database stage (T iterations) and the inter-database stage (T' iterations). At the first stage, we use the progressive learning manner to update the model on a few labeled data $\mathcal{L}$ and the remaining unlabeled source domain data $\mathcal{U}$ for total T iterations. The unlabeled source domain data are pseudo-labeled ($\mathcal{U} \rightarrow \hat{\mathcal{U}}$) and we use the updated model Φ^T_{intra} as an initialization for the inter-database stage. Then, in the inter-database stage, we also update the model by jointly training on the source domain data $\mathcal{S}$ ($\mathcal{S} = \mathcal{L} \cup \hat{\mathcal{U}}$) and the unlabeled target domain training data $\mathcal{T}_{train}$ for total T' iterations.

data cannot guarantee a high and stable performance on the target domain data because the data distributions of them are very different. The classification boundary of a classifier trained the source domain data usually is thus not ideal to classify the data in the target domain.

To address the above issue, we propose a mechanism called adaptive transfer to ameliorate the domain bias and stabilize the inter-database performance by gradually increasing the contribution of unlabeled data from the target domain $\mathcal{T}_{train}$ for training. To be specific, we design a dynamic weight to increase the contribution of the pseudo-labeled data $\mathcal{A}_{t'}$ selected from $\mathcal{T}_{train}$ in accordance with the number of iteration t'. To increase the impact of the target domain data as the iteration increases, we design the objective function as following:

$$L = \sigma_{t'} \cdot L^{\mathcal{T}} + (1 - \sigma_{t'}) \cdot L^{\mathcal{S}}, \tag{4}$$

$$\begin{aligned} L^{\mathcal{S}} = & \sum_{(x_i, y_i) \in \mathcal{L}} L_{CE}(\Phi^{t'}_{inter}(x_i), y_i) \\ & + \sum_{(x_i, \hat{y_i}) \in \hat{\mathcal{U}}} L_{CE}(\Phi^{t'}_{inter}(x_i), \hat{y_i}), \end{aligned} \tag{5}$$

$$L^{\mathcal{T}} = \sum_{(x_j, \hat{y_j}) \in \mathcal{A}_{t'}} L_{CE}(\Phi^{t'}_{inter}(x_j), \hat{y_j}). \tag{6}$$

where L_{CE} denotes the cross entropy loss, $\sigma_{t'}$ is the dynamic weight depending on the iteration number t'. The equation of our adaptive transfer module has a similar format with the momentum updating mechanism in [29]. However, the momentum coefficient has a fixed value while our dynamic weight changes according to the iteration number. Besides, our adaptive module aims to increase the impact of the selected and pseudo-labeled target domain data, but [29] targets for making the parameters of the key encoder evolve smoothly. Specifically, the dynamic weights for different datasets are all line with the selecting ratio ($\sigma_{t'}$) depending on the iteration number t'. Since the dynamic weight $\sigma_{t'}$ increases from 0% to 100% as the iteration progresses, the focus of the model optimization gradually is transferred from the source domain to target domain. Particularly, $\sigma_0 = 0\%$ indicates that the CNN model optimization totally employs the source domain data. At the last iteration $t' = T'$, $\sigma_{T'} = 100\%$ demonstrates that the CNN model optimization completely depends on the pseudo-labeled data from the target domain.

IV. Experiments

We conduct extensive experiments in both the intra-database and inter-database scenarios to demonstrate the effectiveness of our method. In this section, we firstly introduce the employed datasets (Sec. IV-A), and then the evaluation metrics (Sec. IV-B). We also describe our implementation details (Sec. IV-C) and demonstrate the experimental results.

A. Databases

Four public databases CASIA-FASD [30] (denoted as C), Idiap Replay-Attack [15] (denoted as I), OULU-NPU [31] (denoted as O), MSU-MFSD [32] (denoted as M) are utilized to evaluate our method in both intra-database and inter-database scenarios. CASIA-FASD consists of 50 subjects and each subject has 12 videos with different resolutions and illumination conditions. Idiap Replay-Attack has in total 1,300 videos for 50 subjects. OULU-NPU contains 3,960 spoofing face videos and 990 living face videos. MSU-MFSD consists of 280 video clips recorded from 35 subjects. CASIA-FASD, Replay-Attack and MSU-MFSD contain low-resolution videos, while OULU-NPU is a large-scale high resolution database. According to the standard protocols of data division, CASIA-FASD consists of 45,014 and 65,381 spoofing faces as the training set and testing set, respectively. Replay-Attack has 92,457 face images in the training set and 122,074 images in the testing set. MSU-MFSD has a training set of 33,574 face images and a testing set of 44,242 faces. OULU-NPU contains 120,922 faces in the training set and 121,175 ones in the testing set.

In the inter-database scenario, we conduct experiments on four testing tasks to verify the generalization of our method to unseen spoofing attacks following previous work [33], [5], [32]. We randomly select one of the databases as the target domain for testing and the other three as the training dataset. Thus, we have four testing scenarios O&C&I to M, O&M&I to C, O&C&M to I and I&C&M to O. For example, O&C&I to M represents that OULU-NPU (O), CASIA-FASD (C), Idiap Replay-Attack (I) are exploited as the source domain data and MSU-MFSD (M) is used as the target domain data.

B. Evaluation Metrics

The Equal Error Rate (EER) evaluation metric is employed in the intra-database testing. EER is the error rate of a verification system when a threshold for the accept/reject decision is chosen such that the probabilities of false acceptance and false rejection are equal. In the inter-database testing, we evaluate all the methods by Half Total Error Rate (HTER). HTER is the half of the sum of the False Rejection Rate (FRR) and the False Acceptance Rate (FAR). FAR is the ratio between FP and the total number of spoofing attacks, and FRR is the ratio between FN and the total number of living faces. To compute HTER, we first compute EER on the target domain data and then a global threshold corresponding to the one used in EER is applied. In addition, Area Under Curve (AUC) is also used to evaluate the overall classification performance.

C. Implementation Details

A typical network ResNet-18 [34] pretrained on ImageNet [35] is employed as our backbone network. For the face data without bounding boxes annotations, we detect the face regions using MTCNN algorithm [36]. All the detected faces are resized to 256×256 as the input of the framework.

We employ SGD optimizer to update our model, where the weight decay is set to 5e-4 and momentum is set to 0.9. In each iteration step of our progressive learning method, we train our model for 20 epochs. Moreover, we set the threshold μ for the prediction confidence score to 0.9. We select $S = 50$ samples from the training set (in the intra-database case) or each source domain (in the inter-database case) as the labeled subset in the experiments. We also evaluate testing performance with respect to different values of S in Sec. IV-F4.

D. Evaluation Results in Intra-database Scenario

We conduct the experiments in the intra-database scenario on four datasets, respectively. There are four testing protocols for OULU-NPU. Protocol 1 is to evaluate the generalization of methods under previously unseen illumination scene. Protocol 2 aims to evaluate the effect of attacks created with different recording devices (e.g., printers or displays). Protocol 3 utilizes a Leave One Camera Out protocol, in order to study the effect of the input camera variation. Protocol 4 considers all the above factors and integrates all the constraints from protocols 1 to 3, therefore protocol 4 is the most challenging. Our method requires much fewer labeled data compared to other fully-supervised methods. We randomly select 50 labeled data from the training set as the initial labeled subset, and then progressively assign pseudo labels to the remaining unlabeled training data. We report the results of using only 50 labeled faces at the bottom of Table I. Using only 50 labeled samples, the CNN baseline S-CNN suffers obvious performance degradation compared to the one using the entire labeled training set, *i.e.*, EER increases from 0.28% (using ninety thousand labeled samples) to 15.63% (using 50 labeled samples) on Replay-Attack, from 0.53% (using forty-five thousand labeled samples) to 12.25% (using 50 labeled samples) on CASIA-FASD, from 0.18% (using thirty-three thousand labeled samples) to 16.29% (using 50 labeled samples) on MSU-MFSD. Such a few labeled samples lead to unstable evaluation results for the CNN network: results exhibit obvious variance. This experiment demonstrates that a CNN network trained with such a few labeled data easily suffers from overfitting.

TABLE I
COMPARISONS WITH THE STATE-OF-THE-ART METHODS. - REPRESENTS THAT THE VALUE IS NOT PROVIDED IN THE CORRESPONDING PAPER. 'S-CNN' IS A STRONG CNN BASELINE, 'PL' DENOTES OUR PROGRESSIVE LEARNING ALGORITHM, 'TC' IS THE PROPOSED TEMPORAL CONSTRAINT MECHANISM AND TC^{avg} IS THE AVERAGE CONSTRAINT MECHANISM.

Method	REPLAY-ATTACK	CASIA-FASD	MSU-MFSD
	EER(%) ↓	EER (%) ↓	EER (%) ↓
With full labeled samples			
LBP+LDA [11]	18.25	21.01	-
IQA [37]	-	32.46	-
CDD [13]	9.75	11.85	-
IDA [32]	-	12.97	8.58
Patch-CNN [38]	0.72	4.44	-
Color [39]	0.42	2.17	4.9
GFA-CNN [40]	0.30	8.3	7.5
S-CNN	0.28	0.53	0.18
With only 50 labeled samples			
S-CNN	15.63±5.45	12.25±4.22	16.29±6.24
S-CNN+PL	1.93±0.64	3.21±0.79	3.17±0.56
S-CNN+PL+TC^{avg}	1.29±0.81	3.08±0.44	1.08±0.29
S-CNN+PL+TC (**Ours**)	**0.36±0.28**	**0.69±0.39**	**0.64±0.27**

TABLE II
RESULTS OF INTRA-DATABASE TESTING ON OULU-NPU.

Prot.	Method	APCER(%)	BPCER(%)	ACER(%)
1	FaceDs [6]	1.2	1.7	1.5
	STASN [4]	1.2	2.5	1.9
	CDCN [41]	0.4	0.0	0.2
	S-CNN	2.4	1.2	1.8
	S-CNN w/ 50 samples	26.3±10.6	21.3±9.7	24.2±9.2
	S-CNN+PL+TC(Ours) w/ 50 samples	0.6±0.4	0.0±0.0	0.4±0.2
2	FaceDs [6]	4.2	4.4	4.3
	STASN [4]	4.2	0.3	2.2
	CDCN [41]	1.8	0.8	1.3
	S-CNN	3.7	1.5	2.6
	S-CNN w/ 50 samples	24.2±9.8	19.2±8.2	21.7±8.3
	S-CNN+PL+TC(Ours) w/ 50 samples	1.7±0.9	0.6±0.3	1.2±0.5
3	FaceDs [6]	4.0±1.8	3.8±1.2	3.6±1.6
	STASN [4]	4.7±3.9	0.9±1.2	2.8±1.6
	CDCN [41]	1.7±1.5	2.0±1.2	1.8±0.7
	S-CNN	3.3±2.6	2.6±1.4	3.1±1.8
	S-CNN w/ 50 samples	22.1±14.3	17.9±12.7	19.6±12.5
	S-CNN+PL+TC(Ours) w/ 50 samples	1.5±0.9	2.2±1.0	1.7±0.8
4	FaceDs [6]	1.2±6.3	6.1±5.1	5.6±5.7
	STASN [4]	6.7±10.6	8.3±8.4	7.5±4.7
	CDCN [41]	4.2±3.4	5.8±4.9	5.0±2.9
	S-CNN	9.2±7.6	7.5±6.4	8.9±7.0
	S-CNN w/ 50 samples	27.8±22.4	21.2±24.6	26.3±22.9
	S-CNN+PL+TC(Ours) w/ 50 samples	5.2±2.0	4.6±4.1	4.8±2.0

In contrast, with the help of our progressive transfer learning, we obtain competitive results on the three databases by using only 50 labeled training data. Compared to S-CNN trained on 50 samples, our method reduces the EER by 15.27%, 11.56% and 15.65% on Replay-Attack, CASIA-FASD and MSU-MFSD, respectively. Moreover, our method

TABLE III
EVALUATION ON REPLAY-ATTACK WITH DIFFERENT S AND μ.

S	EER (%) ↓	AUC (%) ↑	μ	EER (%) ↓	AUC (%) ↑
10	20.22	90.75	0.80	0.87	98.65
20	4.54	97.96	0.85	0.38	99.08
50	0.38	99.85	0.90	0.35	99.91
100	0.33	99.90	0.95	0.42	99.80
$1‰\times N$	0.40	99.93	0.98	0.94	99.73

achieves high performance with low variance, as indicated by the standard deviation (*i.e.*, only 0.28%, 0.39% and 0.27% on these three databases, respectively). This demonstrates the strong stability of our method. The performance of our method is as the same magnitude as the fully-supervised S-CNN, which can be regarded as the upper bound. In particular, our method obtains 0.36% on EER while a state-of-the-art method GFA-CNN achieves 0.3% on EER on Replay-Attack database. Results in Table II demonstrate that our proposed method performs well on the four testing protocols. Since OULU-NPU mainly aims to assess the generalization performances among different conditions, the S-CNN obtains slightly worse performance, which indicates its poor generalization ability. It is worth noting that our method has a lower std, *e.g.*, 0.8% ACER std (lowest) and 2.0% ACER std (lowest) for protocol 3 and protocol 4 respectively, indicating its good stability.

E. Evaluation Results in Inter-database Scenario

To verify the generalization ability to unseen spoofing attacks, we conduct experiments in four testing scenarios: O&C&I to M, O&M&I to C, O&C&M to I and I&C&M to O. In the four testing scenarios, only one database ($\mathcal{T}$) is chosen as the target domain. In each target domain, it consists of a training subset $\mathcal{T}_{train}$ and a testing subset $\mathcal{T}_{test}$. The remaining three databases are used as the source domains, and they have N_1, N_2, N_3 face images respectively. For example, in the testing scenario of O&C&I to M, O&C&I are the source domain databases and M is the target domain database. In our progressive learning algorithm, we randomly select S samples ($S \ll N_1$, $S \ll N_2$, $S \ll N_3$) from each source domain to constitute the initial labeled subset $\mathcal{L}$. The remaining unlabeled data of the source domains are denoted as $\mathcal{U}$.

Firstly, we utilize the labeled data to update the initial CNN model $\Phi^0_{intra}(\cdot)$. After updating over T iterations in the intra-database scenario, all the unlabeled source domain data are pseudo labeled and then incorporated into the training data. Then, the model $\Phi^T_{intra}(\cdot)$ is used to initialize the model $\Phi^0_{inter}(\cdot)$ for the inter-database scenario. In each iteration of the inter-database scenario, we also utilize the updated model to estimate pseudo labels for the unlabeled target domain training data $\mathcal{T}_{train}$. The model continues to update in the inter-database scenario until all the unlabeled target domain training data $\mathcal{T}_{train}$ are pseudo-labeled. Finally, we evaluate the model $\Phi^{T'}_{inter}(\cdot)$ on the target domain testing data $\mathcal{T}_{test}$.

To prove the effectiveness of our progressive transfer learning algorithm in the inter-database scenario, we conduct experiments on two settings: (i) "all data from source domains are labeled" and (ii) "only 50 faces from each source domain are labeled". Note that, case (i) can be regarded as the upper bound of case (ii). We report evaluation results of the two cases in Table. IV. In the first case, the proposed method utilizes the entire labels of all source domains to train the model like other fully-supervised methods, and then we use the trained model to gradually exploit the unlabeled target domain training data. The evaluation results on the testing set of the target domain are shown in the middle part of Table. IV.

As indicated by the experimental results in Table. IV, our method achieves state-of-the-art performance with lower variances in case (i). For instance, our method surpasses the state-of-the-art method SSDG-R* by 7.13% HTER and 3.3% AUC on O&M&I to C, and also reduces the standard deviations from 1.24% to 0.81% (on HTER) and 0.94% to 0.77% (on AUC). Note that, we reproduce the results of SSDG-R with its official code, denoted as SSDG-R*. The reason is that SSDG-R evaluates its model using only two frames randomly selected from each video segment in the target domain data, while we evaluate all the images in the testing set of the target domain. Thus, for fair comparisons, we reproduce the results of SSDG-R using the same training and testing data as ours.

In the second case, we conduct extensive experiments using only 50 labeled samples from each source domain. The experimental results demonstrate that our method achieves superior performance compared to the other fully-supervised methods when only a few labeled samples are available. Specifically, the CNN baseline S-CNN performs poorly with a higher variance compared to our proposed method using only 50 labeled face images. Moreover, our method is still able to attain competitive results compared to our model trained with the entire labels of the source domains, *e.g.*, 10.73% versus 7.82% on HTER in the case O&C&I to M.

F. Ablation Studies

In this section, we investigate the impact of each design in our proposed progressive transfer learning method.

1) ***Progressive learning algorithms (PL):*** According to the results in Table. I and Table. IV, our progressive transfer learning algorithms improve the anti-spoofing performance significantly. For example, using PL, we reduce HTER from 15.63% to 1.93% on Replay-Attack, from 12.25% to 3.21% on CASIA-FASD, and from 16.29% to 3.17% on MSU-MFSD in the intra-database scenario with 50 labeled samples.

In the experiments of the inter-database scenarios, we also observe that the most obvious improvements come from our progressive learning fashion. Especially with only a few (*i.e.*,∼50) labeled data, the CNN baseline S-CNN equipped with the progressive learning algorithm (PL) obtains improvements on HTER by 6.72%, 21.96%, 5.48% and 12.2% respectively in the four inter-database testing scenarios. Therefore, our progressive learning fashion is able to handle the FAS problem with only a few labeled data, significantly reducing the cost of human annotations.

2) ***Temporal Constraint (TC):*** We investigate the impacts of TC on pseudo label accuracy on three databases in Figure. 4. As shown in Figure. 4, employing TC results in a robust pseudo label accuracy. In contrast, the accuracy of pseudo label is much inferior when TC is not applied. If unreliable pseudo

TABLE IV
COMPARISON RESULTS BETWEEN THE PROPOSED METHOD AND STATE-OF-THE-ART METHODS IN THE INTER-DATABASE SCENARIO.

Method	O&C&I to M		O&M&I to C		O&C&M to I		I&C&M to O	
	HTER (%) ↓	AUC (%) ↑	HTER (%) ↓	AUC (%) ↑	HTER (%) ↓	AUC (%) ↑	HTER (%) ↓	AUC (%) ↑
All data from source domains are labeled								
Color Texture [39]	28.09	78.47	30.58	76.89	40.40	62.78	63.59	32.71
MADDG [33]	17.69	88.06	24.50	84.51	22.19	84.99	27.89	80.02
SSDG-R [42]	7.38	97.17	10.44	95.94	11.71	96.59	15.61	91.54
SSDG-R*	15.81±2.50	90.44±1.94	11.14±1.24	95.66±0.94	23.61±1.54	79.30±3.07	21.82±1.25	85.95±1.12
S-CNN	16.52±3.10	91.21±2.77	14.69±2.95	94.10±2.03	28.75±1.20	78.87±4.12	20.48±4.47	87.47±1.01
S-CNN+PL	12.83±3.65	92.07±3.19	10.11±3.04	95.18±2.66	19.55±3.28	88.82±4.52	18.04±3.58	88.91±2.13
S-CNN+PL+TC	12.31±2.73	94.63±2.85	7.85±2.24	97.34±1.99	16.97±2.45	91.67±4.29	16.61±3.10	90.56±2.76
S-CNN+PL+AT	9.67±1.55	95.81±1.25	7.32±1.62	97.78±0.59	14.27±1.03	95.21±2.19	16.18±1.40	92.45±1.74
S-CNN+PL+TC+AT (Ours)	**7.82±1.21**	**97.67±1.09**	**4.01±0.81**	**98.96±0.77**	**10.36±1.86**	**97.16±1.04**	**14.23±0.98**	**93.66±0.75**
Only 50 faces from each source domain are labeled								
S-CNN	21.70±5.88	86.74±6.36	37.71±7.12	68.44±8.45	29.32±4.32	72.46±5.18	32.18±5.23	73.93±6.52
S-CNN+PL	14.98±5.02	91.54±4.71	15.75±4.87	93.76±4.21	23.84±5.79	80.02±4.52	19.98±5.01	86.09±5.58
S-CNN+PL+TC	14.12±4.15	91.65±4.28	12.75±3.76	94.01±4.42	19.65±4.24	83.35±2.86	19.74±3.55	88.34±3.89
S-CNN+PL+AT	11.89±3.52	94.23±3.15	8.63±4.29	94.93±3.23	15.54±3.10	87.67±3.13	16.01±2.36	89.97±1.79
S-CNN+PL+TC+AT (Ours)	**10.73±3.76**	**96.56±2.67**	**6.51±3.22**	**96.12±3.34**	**14.89±2.39**	**93.11±2.15**	**15.73±1.97**	**91.96±1.43**

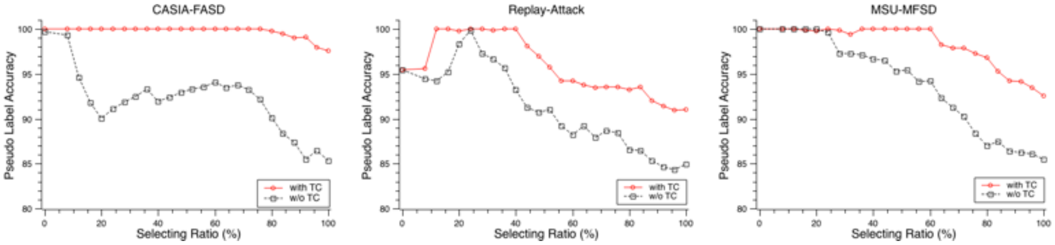

Fig. 4. The impact of the designed temporal constraint mechanism on the accuracy of predicted pseudo labels in the intra-database scenario.

labels are incorporated into the training data, the classifier of the model would fail to learn a correct classification boundary and the model would in return select more erroneously pseudo-labeled data. In this way, the model will fail to detect spoofing faces. Moreover, we also report the results with and without TC in the intra-database scenario and inter-base scenario in Table I and Table. IV, respectively. TC reduces EER by 1.57%, 2.52% and 2.53% on Replay-Attack, CASIA-FASD and MSU-MFSD respectively. We also demonstrate the results of using the average TC, denoted as TC^{avg}, in Table I. The average TC takes the averaged prediction confidence score $\bar{p}$ as the final confidence score for all frames. When we use the average TC as the confidence score, faces from the same video segment will have the same confidence score. Since we select pseudo-labeled faces according to the prediction confidence scores, faces from the same video might be often selected. The diversity of the selected face images will decrease, and insufficiently diverse training data will handicap learning a discriminative classification boundary. Therefore, the results of TC^{avg} are inferior to that with our TC.

In the inter-database testing experiments indicated by Table. IV, employing TC reduces 1.16%, 2.12%, 0.65% and 0.28% on HTER and obtains 2.33%, 1.19%, 5.44% and 1.99% improvements on AUC in the four inter-database scenarios. Furthermore, the results of using TC also exhibit lower variances, demonstrating TC also improves the robustness.

*3) **Adaptive Transfer (AT):*** The scale of the selected data subset continues to develop along with the increasing iteration number in our progressive transfer learning. Thus, it is thoughtful to gradually increase the contribution of the unlabeled data from the target domain for training. In our method, we design a dynamic weight to increase the contribution of the target domain pseudo-labeled data in accordance with the number of iteration. The experimental results with AT are shown in Table. IV that the CNN model not using AT obtains worse performance than the model trained with AT. Specifically, AT reduces 3.39%, 6.24%, 4.76% and 4.01% on HTER in the four inter-database scenarios, respectively. In addition, the standard deviations of the inter-database testing results are lessened when AT is used. These results illustrate that AT ameliorates the domain bias and further improves the robustness of our model in the inter-database scenario.

*4) **Impact of the number of initially labeled data** S:* In both the intra-database scenario and the inter-database scenario, our progressive transfer learning algorithm randomly selects a few labeled data as the initially labeled set at the first iteration. To study the impact of different values of S, we set $S \in (20, 30, 40, 50, \mathcal{N})$ and $\mathcal{N}$=1‰$\times N$. In Table. III, we observe that only 50 samples are enough to attain competitive performance, and more human-annotated labels only bring slight improvements. Thus, we set S=50 in our experiments. Compared with the fully-supervised methods that usually use a large number of labeled spoofing data, the proposed method greatly relaxes the tediousness data labeling.

*5) **Impact of the threshold** μ:* At each learning iteration, we select the unlabeled data with high-confident pseudo labels and incorporate them into the training data. In the data selection process, we filter out low-confident data whose prediction confidence score is less than the threshold μ, which is utilized to verify the reliability of pseudo labels of the

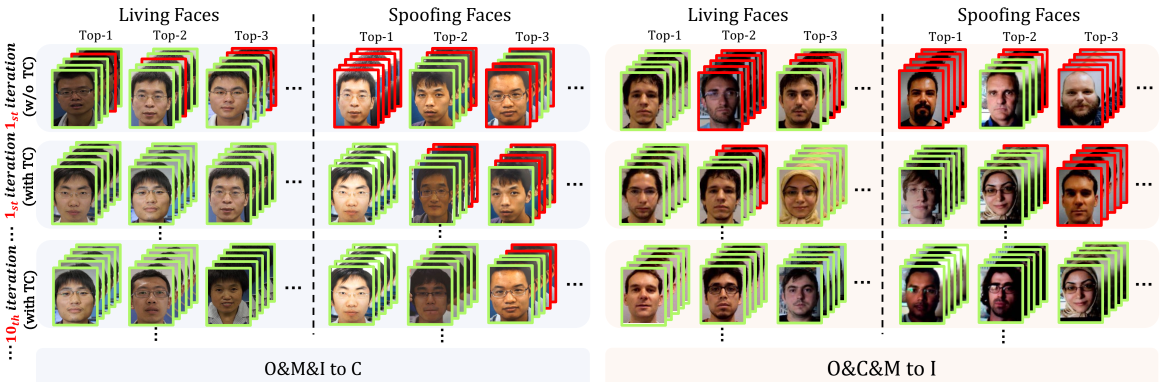

Fig. 5. Visualization results under different experimental conditions of O&M&I to C and O&C&M to I. Top-3 person results are shown. Here, we only illustrate five pictures for each person. The accurate pseudo-label predictions are displayed in green boxes, and the inaccurate ones are displayed in red boxes.

selected data. We experiment with different values of the threshold μ ($\mu \in \{0.80, 0.85, 0.90, 0.95, 0.98\}$) to test whether our algorithm is sensitive to μ. As shown in Table. III, using different μ, the performance of our method only changes slightly. This might be due to the selected data with highly-confident pseudo labels always have high confidence score. Note that using a much higher threshold (*i.e.*, 0.98), our method might struggle to find sufficient examples while using a lower threshold (*i.e.*, 0.80), our method may select some data with erroneous pseudo-labels. However, our method is overall insensitive to the choice of μ. In all experiments, we set $\mu = 0.90$ to ensure the reliability of the pseudo labels.

V. Visualization

To better understand the selected data during our progressive transfer learning and the improvements brought by the TC mechanism, we visualize the selected data in different iterations in Figure. 5. As visible in the visualization results of these two testing scenarios (O&M&I to C and O&C&M to I), we observe that spoofing faces are easily misidentified as the living faces. This indicates that the CNN model is prone to producing high confidence on the living faces of the target domain, whereas the spoofing faces of the target domain are difficult for the model trained on the source domains to distinguish. Moreover, we find that the designed TC mechanism improves the pseudo label accuracy by comparing the faces in the second row with the faces in the first row. Our progressive learning method also refines the results as our iteration progresses. Note that, less inaccurate pseudo labels occur at 10-th iterations, as shown in the third row of Figure 5.

VI. Conclusion

In this paper, we proposed a progressive transfer learning method for FAS to tackle face spoofing attacks with only a few labeled training data. To be specific, by progressively incorporating unlabeled data into our training dataset and our presented temporal consistency constraint, we obtain more reliable pseudo-labeled data to update our model and thus achieve comparable anti-spoofing performance to the state-of-the-art fully supervised methods. Furthermore, our method can be easily extended to handle new types of unlabeled spoofing data with our proposed adaptive transfer mechanism due to its unsupervised and online updating nature. After the model update, our model achieves better anti-spoofing performance on addressing new types of spoofing attacks in comparison to the state-of-the-art. This demonstrates that our method is more practical for real-world application scenarios.

References

[1] Z. Huang, S. Shan, R. Wang, H. Zhang, S. Lao, A. Kuerban, and X. Chen, “A benchmark and comparative study of video-based face recognition on cox face database,” *IEEE Transactions on Image Processing*, vol. 24, no. 12, pp. 5967–5981, 2015.

[2] X. Yin and X. Liu, “Multi-task convolutional neural network for pose-invariant face recognition,” *IEEE Transactions on Image Processing*, vol. 27, no. 2, pp. 964–975, 2017.

[3] S. Zhang, X. Wang, A. Liu, C. Zhao, J. Wan, S. Escalera, H. Shi, Z. Wang, and S. Z. Li, “A dataset and benchmark for large-scale multi-modal face anti-spoofing,” in *CVPR*, 2019, pp. 919–928.

[4] X. Yang, W. Luo, L. Bao, Y. Gao, D. Gong, S. Zheng, Z. Li, and W. Liu, “Face anti-spoofing: Model matters, so does data,” in *CVPR*, 2019, pp. 3507–3516.

[5] Y. Liu, A. Jourabloo, and X. Liu, “Learning deep models for face anti-spoofing: Binary or auxiliary supervision,” in *CVPR*, 2018, pp. 389–398.

[6] A. Jourabloo, Y. Liu, and X. Liu, “Face de-spoofing: Anti-spoofing via noise modeling,” in *ECCV*, 2018, pp. 290–306.

[7] K. Kollreider, H. Fronthaler, M. I. Faraj, and J. Bigun, “Real-time face detection and motion analysis with application in “liveness” assessment,” *IEEE Transactions on Information Forensics and Security*, vol. 2, no. 3, pp. 548–558, 2007.

[8] G. Pan, L. Sun, Z. Wu, and S. Lao, “Eyeblink-based anti-spoofing in face recognition from a generic webcamera,” in *ICCV*, 2007, pp. 1–8.

[9] S. Bharadwaj, T. I. Dhamecha, M. Vatsa, and R. Singh, “Computationally efficient face spoofing detection with motion magnification,” in *CVPR-W*, 2013, pp. 105–110.

[10] W. Kim, S. Suh, and J.-J. Han, “Face liveness detection from a single image via diffusion speed model,” *IEEE transactions on Image processing*, vol. 24, no. 8, pp. 2456–2465, 2015.

[11] T. de Freitas Pereira, A. Anjos, J. M. De Martino, and S. Marcel, “Can face anti-spoofing countermeasures work in a real world scenario?” in *ICB*, 2013, pp. 1–8.

[12] J. Komulainen, A. Hadid, and M. Pietikäinen, “Context based face anti-spoofing,” in *BTAS*, 2013, pp. 1–8.

[13] J. Yang, Z. Lei, S. Liao, and S. Z. Li, “Face liveness detection with component dependent descriptor,” in *ICB*, 2013, pp. 1–6.

[14] J. Määttä, A. Hadid, and M. Pietikäinen, “Face spoofing detection from single images using texture and local shape analysis,” *IET biometrics*, vol. 1, no. 1, pp. 3–10, 2012.

[15] I. Chingovska, A. Anjos, and S. Marcel, “On the effectiveness of local binary patterns in face anti-spoofing,” in *BIOSIG*, 2012, pp. 1–7.

[16] A. Pinto, H. Pedrini, W. R. Schwartz, and A. Rocha, “Face spoofing detection through visual codebooks of spectral temporal cubes,” *IEEE Transactions on Image Processing*, vol. 24, no. 12, pp. 4726–4740, 2015.

[17] Y. Liu, J. Stehouwer, A. Jourabloo, and X. Liu, “Deep tree learning for zero-shot face anti-spoofing,” in *CVPR*, 2019, pp. 4680–4689.

[18] C. Gan, Z. Gan, X. He, J. Gao, and L. Deng, “Stylenet: Generating attractive visual captions with styles,” in *CVPR*, 2017, pp. 3137–3146.

[19] Y. Bengio, J. Louradour, R. Collobert, and J. Weston, “Curriculum learning,” in *ICML*, 2009, pp. 41–48.

[20] M. P. Kumar, B. Packer, and D. Koller, “Self-paced learning for latent variable models,” in *NeurIPS*, 2010, pp. 1189–1197.

[21] F. Khan, B. Mutlu, and J. Zhu, “How do humans teach: On curriculum learning and teaching dimension,” in *NeurIPS*, 2011, pp. 1449–1457.

[22] C. Gong, D. Tao, S. J. Maybank, W. Liu, G. Kang, and J. Yang, “Multi-modal curriculum learning for semi-supervised image classification,” *IEEE Transactions on Image Processing*, vol. 25, no. 7, pp. 3249–3260, 2016.

[23] A. Graves, M. G. Bellemare, J. Menick, R. Munos, and K. Kavukcuoglu, “Automated curriculum learning for neural networks,” in *ICML*, 2017, pp. 1311–1320.

[24] C. Gan, Y. Li, H. Li, C. Sun, and B. Gong, “Vqs: Linking segmentations to questions and answers for supervised attention in vqa and question-focused semantic segmentation,” in *ICCV*, 2017, pp. 1811–1820.

[25] Y. Wu, Y. Lin, X. Dong, Y. Yan, W. Bian, and Y. Yang, “Progressive learning for person re-identification with one example,” *IEEE Transactions on Image Processing*, vol. 28, no. 6, pp. 2872–2881, 2019.

[26] Y. Ding, H. Fan, M. Xu, and Y. Yang, “Adaptive exploration for unsupervised person re-identification,” *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, vol. 16, no. 1, pp. 3:1–3:19, 2020.

[27] C. Gan, M. Lin, Y. Yang, Y. Zhuang, and A. G. Hauptmann, “Exploring semantic inter-class relationships (sir) for zero-shot action recognition,” in *AAAI*, vol. 29, no. 1, 2015.

[28] C. Gan, M. Lin, Y. Yang, G. Melo, and A. G. Hauptmann, “Concepts not alone: Exploring pairwise relationships for zero-shot video activity recognition,” in *AAAI*, vol. 30, no. 1, 2016.

[29] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick, “Momentum contrast for unsupervised visual representation learning,” in *CVPR*, 2020, pp. 9729–9738.

[30] Z. Zhang, J. Yan, S. Liu, Z. Lei, D. Yi, and S. Z. Li, “A face antispoofing database with diverse attacks,” in *ICB*, 2012, pp. 26–31.

[31] Z. Boulkenafet, J. Komulainen, L. Li, X. Feng, and A. Hadid, “Oulu-npu: A mobile face presentation attack database with real-world variations,” in *FG*, 2017, pp. 612–618.

[32] D. Wen, H. Han, and A. K. Jain, “Face spoof detection with image distortion analysis,” *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 4, pp. 746–761, 2015.

[33] R. Shao, X. Lan, J. Li, and P. C. Yuen, “Multi-adversarial discriminative deep domain generalization for face presentation attack detection,” in *CVPR*, 2019, pp. 10 023–10 031.

[34] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *CVPR*, 2016, pp. 770–778.

[35] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, “Imagenet: A large-scale hierarchical image database,” in *CVPR*, 2009, pp. 248–255.

[36] K. Zhang, Z. Zhang, Z. Li, and Y. Qiao, “Joint face detection and alignment using multitask cascaded convolutional networks,” *IEEE Signal Processing Letters*, vol. 23, no. 10, pp. 1499–1503, 2016.

[37] J. Galbally and S. Marcel, “Face anti-spoofing based on general image quality assessment,” in *ICPR*, 2014, pp. 1173–1178.

[38] Y. Atoum, Y. Liu, A. Jourabloo, and X. Liu, “Face anti-spoofing using patch and depth-based cnns,” in *IJCB*, 2017, pp. 319–328.

[39] Z. Boulkenafet, J. Komulainen, and A. Hadid, “Face spoofing detection using colour texture analysis,” *IEEE Transactions on Information Forensics and Security*, vol. 11, no. 8, pp. 1818–1830, 2016.

[40] X. Tu, Z. Ma, J. Zhao, G. Du, M. Xie, and J. Feng, “Learning generalizable and identity-discriminative representations for face anti-spoofing,” *ACM Transactions on Intelligent Systems and Technology*, vol. 11, no. 5, pp. 1–19, 2020.

[41] Z. Yu, C. Zhao, Z. Wang, Y. Qin, Z. Su, X. Li, F. Zhou, and G. Zhao, “Searching central difference convolutional networks for face anti-spoofing,” in *CVPR*, 2020, pp. 5295–5305.

[42] Y. Jia, J. Zhang, S. Shan, and X. Chen, “Single-side domain generalization for face anti-spoofing,” in *CVPR*, 2020, pp. 8484–8493.