Machine Intelligence Research
www.mi-research.net

DOI: 10.1007/s11633-022-1341-4

FedFV: A Personalized Federated Learning Framework for Finger Vein Authentication

Feng-Zhao Lian[1,2] Jun-Duan Huang[1] Ji-Xin Liu[3] Guang Chen[1,2]
Jun-Hong Zhao[1] Wen-Xiong Kang[1]

[1] School of Automation Science and Engineering, South China University of Technology, Guangzhou 510641, China

[2] GRG Banking Equipment Co. Ltd., Guangzhou 510663, China

[3] School of Automation, Guangdong University of Petrochemical Technology, Maoming 525000, China

Abstract: Most finger vein authentication systems suffer from the problem of small sample size. However, the data augmentation can alleviate this problem to a certain extent but did not fundamentally solve the problem of category diversity. So the researchers resort to pre-training or multi-source data joint training methods, but these methods will lead to the problem of user privacy leakage. In view of the above issues, this paper proposes a federated learning-based finger vein authentication framework (FedFV) to solve the problem of small sample size and category diversity while protecting user privacy. Through training under FedFV, each client can share the knowledge learned from its user's finger vein data with the federated client without causing template leaks. In addition, we further propose an efficient personalized federated aggregation algorithm, named federated weighted proportion reduction (FedWPR), to tackle the problem of non-independent identically distribution caused by client diversity, thus achieving the best performance for each client. To thoroughly evaluate the effectiveness of FedFV, comprehensive experiments are conducted on nine publicly available finger vein datasets. Experimental results show that FedFV can improve the performance of the finger vein authentication system without directly using other client data. To the best of our knowledge, FedFV is the first personalized federated finger vein authentication framework, which has some reference value for subsequent biometric privacy protection research.

Keywords: Finger vein, personalized federated learning, privacy protection, biometric, authentication.

Citation: F. Z. Lian, J. D. Huang, J. X. Liu, G. Chen, J. H. Zhao, W. X. Kang. FedFV: A personalized federated learning framework for finger vein authentication. *Machine Intelligence Research*. http://doi.org/10.1007/s11633-022-1341-4

1 Introduction

Finger vein (FV) is an emerging biometric modality that has attracted considerable attention recently[1]. Unlike other biometric modalities[2–4], such as the face, iris, fingerprint, etc., FV has some apparent advantages. 1) FVs are located in the subcutaneous layer of the skin and can only be captured with near-infrared light and a corresponding camera. This unique imaging mechanism can reduce the possibility of FV image theft and spoofing. 2) FV acquisition is contactless (compared to fingerprint) and user-friendly (compared to iris), which is both convenient and beneficial to the user's hygiene.

Due to the advantages aforementioned, FV authentication has a broad application prospect in daily life or industry and has received increasing attention in the field of biometrics. Earlier research on FV authentication mainly based on feature-engineering methods, which can be divided into three categories: Vein pattern-based methods, texture-based methods, and minutiae-based methods[5]. With the rapid growth of computing power, deep convolutional neural networks (CNNs) demonstrate outstanding performance in image understanding and recognition, which have been introduced to the FV image feature extraction and obtained the excellent performance.

However, with the increasing concern about personal privacy data, one of the critical challenges facing deep learning-based FV authentication systems is the difficulty in obtaining sufficient FV images to train models. Although a large number of FV images are available to different clients, such as schools, enterprises, etc., these images cannot be publicly available or shared among these institutions due to potential privacy leakage issues. In the case of FV authentication systems with few training samples, it is challenging to train deep models with satisfactory performance and robustness. Hence, it is significant to develop a framework that combines most clients to train a high-performance model and prevent their FV data from being stolen. Fortunately, we found an effective approach that has been proposed to solve a similar problem: federated learning. Federated learning is a

Research Article
Manuscript received April 17, 2022; accepted May 27, 2022
Recommended by Guest Editor Zhe-Nan Sun

privacy-preserving machine learning technique that can train the deep model decentralized. Therefore, the data from each client can be prevented from being exposed to other clients. Specifically, during the federated learning process, clients will only send the training model parameters to the server, rather than the FV images, which reduces the risk of privacy leakage.

By introducing and improving federated learning, we propose the personalized federated learning framework for FV authentication, abbreviated as FedFV, to better train the model and solve data island and privacy leakage problems. FedFV enables each client to learn from the other participants and achieve better performances than only using local training data. Furthermore, this work also proposes an improved aggregation algorithm to optimize the performance of FedFV. To the best of our knowledge, this is the first work to use federated learning to solve the problem of privacy protection in FV authentication, which can provide some reference value for academic research or practical applications.

The main contributions of this paper can be summarized as follows.

1) The first personalized federated learning framework for FV authentication, namely FedFV, is proposed to address data island and privacy protection issues.

2) A newly personalized aggregation algorithm, namely FedWPR, is proposed to solve the decline of model performance caused by the strong heterogeneity of FV datasets.

3) To simulate real-world scenarios and evaluate the performance of the proposed FedFV, nine public FV datasets are used to conduct the experiments, which can provide extensive comparative data for researchers in this field.

2 Related works

2.1 Finger vein biometrics

Research on FV authentication can be mainly divided into feature engineering-based methods and deep learning-based methods. As pioneers of the feature engineering-based FV authentication method, Miura et al.[6] calculated the local maximum curvature of a cross-section to extract vein pattern-based features. As a binary image of a FV, the vein pattern is often used for template matching. Then, inspired by the maximum curvature algorithm, Song et al.[7] proposed the mean curvature to extract the features of the vein pattern, which is more robust to light intensity. In addition, inspired by Weber′s law, Yang et al.[8] proposed an Alpha-trimmed Weber representation to generate vein pattern features and used the perceptual increment threshold of the human eyes to distinguish the vein region from the background.

Another typical framework, the texture-based method, also extracts the FV feature. Lee et al.[9] used local binary patterns (LBPs) as the texture features of FV images to tackle the problem of different brigthnesses in different areas of FV images. Lu et al.[10] proposed double-orientation coding histogram to tackle the influence of finger rotation. In addition, many other features have been applied to FV authentication, including principal component analysis (PCA) features[11], super pixel features[12], skeleton orientation encoding features[13], and soft biometric features[14].

In recent years, with the development of deep learning, many researchers have explored deep learning-based FV authentication and achieved breakthroughs. To realize real-time FV verification, Fang et al.[15] proposed a lightweight dual-stream neural network, which efficiently tackles the lack of FV datasets. Moreover, Xie and Kumar[16] used a LightCNN network to extract features and introduced a supervised discrete hash to compress features. However, lightweight networks are less robust to finger rotation and offset. Hence, metric learning like a Siamese network[5], triplet loss[17], etc, were adopted to strengthen the feature extraction ability and make the feature more discriminant. In order to increase the features of the FV obtained by the model, Hao et al.[18] proposed a multi-task FV authentication model which integrates region of interest (ROI) interception and feature extraction, and Huang et al.[19] proposed a CNN-based attention model, namely JAFVNet, which can extract the vein texture and finger shape of the raw FV image. In addition, Kuzu et al.[20] used the CNN-LTSM model to process FV image sequences, which allows users to have their finger vein patterns acquired on-the-fly. Yang et al.[21] proposed a multi-output neural network model FV recognition and AntiSpoofing network (FVRAS-Net), which can output authentication results and anti-counterfeit detection results. The most novel and cutting-edge research[22], which is the first to explore the effects of the transformer-based method on FV authentication tasks, proposed the transformer model of FV authentication. This study provides a new idea for FV authentication based on deep learning.

To fully train and further improve deep models′ performance, sufficient data is usually required. However, it is difficult for an organization to obtain a large amount of public data for training. On the other hand, the organization cannot expose its datasets because of privacy protection regulations. In order to tackle the problems above, federated learning is used for FV authentication model training, which can effectively tackle the data island and improve the client model′s performance to protect user privacy.

2.2 Federated learning

Federated learning (FL) was firstly proposed by Google[23], and its core idea is to enable multiple clients to jointly learn a machine learning model without exchanging their local data. FL aggregates all client models to train a global model, which can tackle the problem of data island while protecting user privacy. Google[24, 25] used federated learning to improve Google Keyboard query suggestions. However, the data distribution of the clients is usually based on the specific behavior of users using this client, so each client′s data distribution is likely to be non-independent identically distribution (non-IID), which has always been a challenge for federated learning. Zhao et al[26]. demonstrated that the degree of non-IID is proportional to the performance of global federated learning, which will reduce the accuracy of the model. In order to quantify the influence of non-IID degree on the model, Hsieh et al.[27] designed and evaluated the SkewScout algorithm to reflect the data deviation by adapting to the communication frequency between data partitions. In addition, Zhao et al.[26] calculated the difference in probability distribution between different clients′ data to measure the non-IID degree. To cope with the non-IID challenge, Sahu et al.[28] proposed FedProx to adjust the number of local training epochs according to the dataset of each client. Bai et al.[29] added momentum to the federated training process to recede client drift. Tang et al.[30] resampled the data to weaken the non-IID degree of the data.

These researches mainly tackle the slight non-IID by optimizing the aggregation of the global model. In a real application scenario, different clients collect FV image by different devices in various environments. Obviously, FV images distribution among clients is different. It is almost impossible to train a global model universally suitable for all clients. Therefore, in recent years, personalized federated learning (PFL) has attracted considerable attention from researchers, which can alleviate the problem of non-IID data. PFL is an intermediate paradigm between the server-based FL paradigm that produces a global model and the local model training paradigm[31]. For each client, the model trained by PFL can collect the knowledge of others clients and make the model fit with its own data distribution. Personalized federated learning can be mainly divided into single-model, multi-model, and N-model approaches.

Single-model PFL approaches. For Single-model PFL approaches, the server firstly aggregates the models of all clients into a global FL model. Then, the clients use their own local data to fine-tune the aggregated global FL model. Chen et al.[32] proposed FedHealth that achieved personalized model learning through knowledge transfer. Zhuang et al.[33] proposed federated partial averaging (FedPav), enabling clients that the models are partially different can be federated. In addition, Itahara et al.[34] designed a framework that combined distillation learning with federated learning, which tackled the challenge of non-IID data. To optimize the knowledge transfer between the global model and the local model, Smith et al.[35–37] analyzed the relationship between multi-task learning and federated learning to realize personalized. Moreover, Li et al.[38–41] combined meta-learning with federated learning to solve heterogeneity problems. These researches are based on a global model to realize personalized, which are more suitable for mild non-IID data.

Multi-model PFL approaches. Multi-model PFL is proposed to solve the problem that training a single global model is ineffective if there are inherent partitions among clients or data distributions. Particularly, the server training by multi-model PFL clusters all clients with similar data distribution and the number of aggregation models in the server depended on the number of categories formed by all client clusters. In order to achieve multi-model PFL, there are several excellent kinds of research: Sattler et al.[42] proposed integrating hierarchical clustering into FL as a post-processing step. Huang et al.[43] proposed a community-based FL algorithm to predict patient hospitalization time and mortality.

***N*-model PFL approaches.** For N-model PFL approaches, the server executes a separate set of aggregation algorithms for each client, resulting in N aggregation models in the cloud. N represents the number of all clients, which is different from multi-model PFL. There is also some excellent research about the N-model PFL: Huang et al.[44] calculated the weight in the aggregation process according to the similarity of the dataset. Smith et al.[35] regarded each client as a task in multi-tasking learning. Chai et al.[45] proposed federated learning method with asynchronous tiers (FedAT), a novel, tiered FL framework that updated local model parameters synchronously within the tiers and the global model asynchronously across the tiers.

In the actual application scenario, the devices and environments of FVs collected by different clients are various, which means that the data heterogeneity among clients is unignorable. Hence, the N-model PFL may be the best suitable method for the task of FV authentication.

3 Methodology

This section details the proposed framework of FedFV that applied personalized federated learning to FV authentication. First, the problem definition of FedFV is introduced, which makes it easier to understand the following content. Second, the framework structure of FedFV is introduced, which aims to federate multiple clients to train a better FV authentication system without exposing any client′s local FV data to others. Third, the proposed FedFV-FedWPR is introduced, which is an effective and easy-to-realize algorithm for personalized aggregation. Finally, some assist parts are introduced, including the classifier, loss, and training strategy.

3.1 Problem description

For the deep learning-based FV authentication system, sufficient training samples are significant for per-

single me multiple data distributions handle nhi ho skte

multiple solves that problem by categorising client models of similar data distributions into one categot
and then aggregating over all categories.

formance. However, the amount of FV images owned by different clients is limited and cannot be shared with each other, since it is often restricted by privacy data protection regulations. The latter issue leads to the phenomena called data islands. In this paper, it is assumed that there are N different clients that are denoted as $\{C_1, C_2, C_3, \cdots, C_N\}$. Each institution has its own local FV dataset that is denoted as $\{FV_1, FV_2, FV_3, \cdots, FV_N\}$. Each FV dataset contains M_i subjects, i.e., $FV_i = \{P_1, P_2, P_3, \cdots, P_{M_i}\}$. In the framework of deep learning, the dataset is often divided into the training set $FV_i^{train} = \{P_1, P_2, \cdots, P_{\lfloor 0.8\times M_i \rfloor}\}$ and the test set $FV_i^{test} = \{P_{\lfloor 0.8\times M_i \rfloor+1}, P_{\lfloor 0.8\times M_i \rfloor+2}, \cdots, P_{M_i}\}$. It is worth noting that the authentication task studied in this paper is a subject-independent problem[46], i.e., the subjects of the training set and test set are not overlapped, and each subject is treated as an independent category during training. We use this classification strategy to demonstrate the improved performance of federated learning on model robustness. This task is different from the common image classification task, which has the same categories in the training set and test set. Therefore, traditional FV authentication practice is to distinguish between training sets and test sets, and does not strictly require the use of the validation set.

Then, each client uses its own FV training set FV_i^{train} to train its local model f_i. Due to the insufficient training data, the local model's performance is usually unsatisfactory and difficult to improve. Therefore, in this work, we aim to conduct federated training under the condition of data privacy protection and get N personalized federated models in the server, denoted as $\left\{f_1^{fed}, f_2^{fed}, f_3^{fed}, \cdots, f_N^{fed}\right\}$. Therefore, the objective of FedFV is federated with other clients to achieve performance improvements. Specifically, the equal error rate (EER) is used as the primary metric to measure the performance of the FV authentication model. The mathematical expression of this work's object can be summarized in (1).

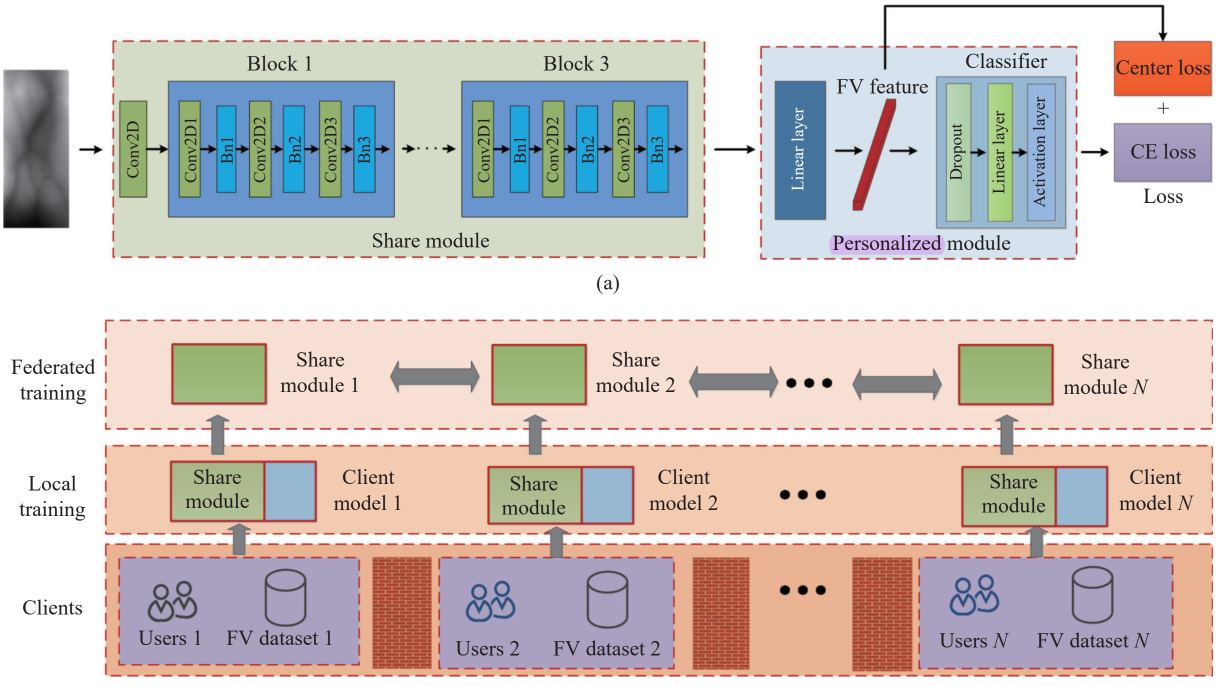
$$EER\left(f_i^{fed}\left(FV_i^{test}\right)\right) < EER\left(f_i\left(FV_i^{test}\right)\right),\ \forall i \in N \qquad (1)$$

where EER denotes equal error rate.

3.2 Framework of FedFV

N MODEL PFL USED

The model architecture of the client is shown in Fig. 1.(a), while the overall framework of the proposed FedFV is shown in Fig 1.(b). The FedFV framework uses N-model personalized federated learning to tackle the non-IID data between clients. In traditional federated learning, the server distributes the initial model to each client, which means that the model structure trained in each client has the same architecture. However, in the FV federated training scenario, different clients have different numbers of users, and the dimension of the classifier in the model depends on the number of users. Hence, keeping the model architecture exactly the same is not



(a)

(b)

Fig. 1 Framework of FedFV. (a) Clients' model, which consists of two parts: The green area represents the shared part, the blue area represents the personalized part; (b) Concise FedFV framework, the clients' models are denoted by the green and blue rectangle. Training each client's model requires two stages: local training for the whole and federated training for the shared part.

optimal. Architectures with significant differences will make it difficult for the server to aggregate each client model. Therefore, in the proposed FedFV, federated partial aggregation[33] is adopted to tackle these problems. In the FedFV framework, the client model is divided into two parts: the shared part and the personalized part. As Fig. 1.(a) shows, the shared part is marked in green, while the personalized part is marked in blue.

During federated training, only the shared parts in clients are uploaded to the server for federated aggregation. The personalized part is kept locally and does not participate in the federated aggregation. Particularly, for the client model, the mobilenetV2[47] is used as the backbone and divided into the shared and personalized parts. The shared part consists of the convolution layer, and the personalized part consists of the linear layer. This is mainly because the convolutional layers of the shared part are used to extract the vein pattern features from the FV image, and such knowledge can usually be shared. In comparison, the linear layers are mainly used for integrating and combining the features extracted in the shared part. Hence, the personalized part remains locally and does not participate in federated aggregation, which is a benefit for retaining personalized knowledge to tackle the non-IID data between clients. The overall process of FedFV is described in Algorithm 1.

In this work, mobilenetV2[47] is adopted as the backbone of the client model due to its excellent performance and lightweight. The framework adopts the N-model-based personalized federated learning stage, which generates a specific aggregation model for each client on the server. The N-model-based method can be represented by (2). After collecting the local models of all clients, the server performs a different model aggregation algorithm for each client. Therefore, when the FV datasets are highly heterogeneous, FedFV can generate a specific aggregation model for each client on the server, making the local model personalized and adapting well to its users.

$$\begin{pmatrix} f_1^{fed} \\ f_2^{fed} \\ \cdots \\ f_N^{fed} \end{pmatrix} = \begin{pmatrix} W_{11} & W_{12} & \cdots & W_{1N} \\ W_{21} & W_{22} & \cdots & W_{2N} \\ \cdots & \cdots & \cdots & \cdots \\ W_{N1} & W_{N2} & \cdots & W_{NN} \end{pmatrix} \begin{pmatrix} f_1 \\ f_2 \\ \cdots \\ f_N \end{pmatrix} = W \times F. \tag{2}$$

agg model = weights*local model

3.3 Federated weighting proportional reduction

The algorithm, which is shown as (2), is commonly used for federating the N-model. The most significant of this algorithm is the design of matrix W, which would be trained easily and improve the performance of each client model as soon as possible. Although there is an excellent algorithm, federated attentive message passing (FedAMP)[44], which designs the aggregation weight matrix w_{ij} that depends on the parameters similarity among each client model. However, FedAMP has to calculate the model parameters, similarity in every round to obtain each client model′s updated weight, which is time-consuming and computationally expensive. In addition, during federated learning, aggregation weights in the server are depended on the scale of the clients′ users. This leads to the problem that if there is a client C with a small number of users, the corresponding aggregation weight will be small too. Hence, the contribution of client C will be minimal in federated learning. The aggregated server model contains very little knowledge of client C, which will not only hinder the client model′s training, but will also prevent the server model from fully learning the various knowledge of the clients.

Algorithm 1. The FedFV training strategy

```
Require: Initial model $f$, $N$ clients′ datasets $\{FV_i\}_{i=1}^N$, $RR$
Ensure: $\left\{f_i^{fed}\right\}_{i=1}^N$
Distribute $f$ to each client;
for round $t$ in range($T$) do
    for client $i$ in range($N$) do
        for epoch $e$ in range($E$) do
            Update the client model $f_i$ with local data $FV_i$;
            Send the client model $f_i$ to server;
        end for
    end for
end for
//*Now the server owns $\{f_i\}_{i=1}^N$;*//
$\left\{f_i^{fed}\right\}_{i=1}^N \leftarrow$ FedWPR;
for client $i$ in range($N$) do
    The server distribute $f_i^{fed}$ to client $i$;
end for
```

Algorithm 2. FedWPR

```
Require: $N$ client local models $\{f_i\}_{i=1}^N$, $N$ clients′ datasets $\{N_i\}_{i=1}^N$, $RR$
Ensure: $\left\{f_i^{fed}\right\}_{i=1}^N$
Initialize $w_1, w_2, w_3, \cdots, w_N$ by (2);
for $i$ in range($N$) do
    $f_i^{fed} = RR \times \sum_{j=1}^N f_j w_j + (1 - RR) f_i$;
end for
```

To tackle both the efficiency and the non-IID problems, a new and effective algorithm is proposed in this work, namely federated weighting proportional reduction (FedWPR). The overall process of FedWPR is described in Algorithm 2. The proposed FedWRR has three advantages. First, FedWRR can accommodate clients with

For every client we will make a seperate aggregation algorithm based on heterogeneous dataset making the model more personalised and well adapted to the particular client.

3 major points about FedWPR

different numbers of users, which is a common and unavoidable situation. Clients with a large number of users, i.e., those who contribute a lot to the server, will have a more significant impact on federated learning. Second, FedWPR can prevent clients with a small number of users from being influenced by clients with a large number of users, which end up burying the personality of their data. Third, to improve training speed, FedWPR is designed to obtain the W-matrix efficiently, so the aggregation process is less computationally intensive while satisfying the performance.

The following described the FedWPR in mathematical terms. First, the server calculates the initial weights based on the user numbers of all clients, as shown in (3).

$$w_k = \frac{n_k}{\sum_{k=1}^{N} n_k} \tag{3}$$

where n_k denotes the dataset size of the kth client. $w^{init} = \{w_1, w_2, w_3, \cdots, w_N\}$ denotes the corresponding initial weight. If w^{init} is directly used to aggregate the local model, the method can be regarded as the FedPav[33] algorithm based on a single model[31]. Due to the non-IID of FV data in different clients, the single model-based method is not the best choice for this task. In addition, for the client model, federated learning increases time and communication costs, which should be matched by performance improvement. Therefore, FedWPR abandoned this way. After obtaining w^{init}, the server sets a parameter to reduce the rate (RR), which will scale the w^{init} in equal proportions. The value of RR is calculated by (6). When calculating the ith client′s personalized federated model f_i^{fed}, the remaining weight (1-RR) is given to the ith client′s local model f_i, which can limit the minimum aggregation weight of f_i. In the aggregation process, the weights w_i^{fed} used in the f_i^{fed} parameter calculation of each personalized federated model are calculated by (4).

$$w_i^{fed} = \{RR \times w_1, RR \times w_i + (1 - RR), \cdots, RR \times w_N\}. \tag{4}$$

The resulting W matrix can be expressed as

$$W = RR \times \begin{pmatrix} 1 & 1 & \cdots & 1 \end{pmatrix}_{N\times 1} \begin{pmatrix} w_1 \\ w_2 \\ \cdots \\ w_N \end{pmatrix}_{1\times N} + \\ (1 - RR)\, I_N \tag{5}$$

$$RR = \frac{1}{2 \times N}. \tag{6}$$

If $RR = 0.9$, the W matrix will be as follows:

$$W = \begin{pmatrix} 0.9w_1 + 0.1 & 0.9w_2 & \cdots & 0.9w_N \\ 0.9w_2 & 0.9w_2 + 0.1 & \cdots & 0.9w_N \\ \cdots & \cdots & \cdots & \cdots \\ 0.9w_1 & 0.9w_2 & \cdots & 0.9w_N + 0.1 \end{pmatrix}. \tag{7}$$

For the federated learning using FedWPR, the first aggregation matrix W is calculated. Then the client gets the training model from the server. It will train with the local FV data in each client. For each training of 1 or 2 epochs, the share parts′ parameters will be sent to the server, where calculates the personalized models using Algorithm 2. Then, the server sends back the different personalized models to its corresponding clients. Each client starts the next round of training. Repeat the above steps until convergence.

3.4 Classifier, loss and training strategy

In this proposed FedFV framework, the structure of the classifier and loss function is shown in Fig. 1.(a). In the local training phase, the client combines the classifier with the loss function to supervise the model to improve the feature extraction ability of the local model. In the inference phase, the classifier and the loss functions are discarded. The FV features are extracted directly from the federate-learned personalized local model and measure the similarity of the features to be authenticated and registered. If it is greater than the preset threshold, it is considered that the two FV images come from the same person; that is, the authentication is successful and vice versa. The loss function is shown in (8).

$$L = L_{CrossEntropyloss} + \alpha L_{CenterLoss} \tag{8}$$

where $L_{CrossEntropyloss}$ denotes the cross-entropy loss[48] and $L_{CenterLoss}$ denotes the center loss[49]. α denotes the weight of center loss, using to balance $L_{CrossEntropyloss}$ and $L_{CenterLoss}$.

4 Experiments

In this section, extensive experiments and their related contents are demonstrated. First, we introduce the nine commonly publicly available FV datasets used for the experiments to simulate the real-world scenarios of FV authentication. Second, the detailed experimental setup and metrics are presented, which help relevant researchers to replicate and compare. Third, the experimental results of our proposed FedFV are shown. Fourth, contrast experiments were conducted to analyze the performance improvement of local models using different collaborator datasets. Fifth, we analyze the performance of federated learning by the number of collaborators. Finally, since there is no research on federated learning-based FV authentication, we compare FedFV with the existing state-of-the-art (SOTA) methods of local training for FV authentication.

4.1 Datasets and evaluation protocol

To simulate the real-world scenarios of FV authentication, nine commonly used public FV datasets were used to conduct experiments aimed at analyzing whether federated learning can solve the data island problem of the FV authentication system. The FV datasets used are MMCBNU-6000[50], HKPU-FV[51], PLUSVein-FV[52], SDUMLA-HMT[53], THU-FVFDT[54], FV-USM[55], UTFVP[56], VERA[57], and SCUT-FV[5]. A brief description of these nine FV datasets, which follow the experimental criteria of [22], is shown in Table 1.

These nine datasets are regarded as nine different clients to simulate the real-world scenarios in which multiple institutions cooperate to train the FV authentication model. For each client, the ratio of training and test sets is 8 : 2 (rounding up the training set according to the method of finding the ceiling and using the remaining as the test set). 80% of the categories are used as the training set and 20% as the test set. The open-set testing is used to prove that federated learning can improve the robustness of the FV authentication model. In the test phase, to simulate the authentication process, two different samples from the same finger in multiple permutations are used to compose the intra-class FV pairs. While the inter-class FV pairs are composed using the two samples from a different finger.

4.2 Experimental details and metrics

In the training phase, the cross-entropy loss is combined with a center loss to supervise the local model. While the cross-entropy loss extends the inter-class distance, the center loss reduces the intra-class distance, making the model more superior in feature clustering. In this way, the feature extraction capability of the local model has been improved. In the inference phase, the client discards the model's classification layer, directly extracts the FV features, and then measures the similarity of the features.

The EER was used as the primary metric in our experiments because it is commonly used in the field of biometric authentication. EER is produced when the false acceptance rate (FAR) and the true acceptance rate (TAR) are equal. If the two FV images are of different classes, but are mistaken for the same class by the system, it is a false acceptance pair. FAR represents the percentage of false acceptance pairs in all inter-class pairs. If the two FV images are of the same classes but are mistaken for the different classes by the system, it is a false rejection pair. FRR represents the percentage of false rejection pairs in all intra-class pairs. The calculations of FAR and FRR are shown in (9) and (10). In addition, we also use the TAR when FAR is 0.01, abbreviated as TAR@FAR = 0.01 metric, which can be a more intuitive reference for practical federated learning applications. The TAR can be calculated as one minus FRR.

$$FAR = \frac{\text{Number of matching scores in false acceptance}}{\text{Number of matching scores}} \tag{9}$$

$$FRR = \frac{\text{Number of matching scores in false rejection}}{\text{Number of matching scores}}. \tag{10}$$

In addition, these experiments use nine datasets as nine clients that analyze the adoption of federated learning to optimize the local FV authentication model. Hence, the weighted average values of their EER and TAR@FAR = 0.01 are also used to verify the overall performance of all participating federated learning clients. The calculation of the weighted averages of the metrics is shown in (11), where number of pairs (NoP) represents the number of authentication pairs, as shown in Table 1, and metric represents the EER metric or the TAR@FAR = 0.01 metric.

Table 1 Summary of the experimental FV datasets

FV datasets	Fingers	Captures times	Total images	Training set	Test set	Authentication pairs
HKPU-FV	312	6	1 872	1 500	372	1 860
MMCBNU-6 000	600	10	6 000	4 800	1 200	10 800
PLUSVein-FV	360	5	1 800	1 440	360	1 440
SDUMLA-HMT	636	6	3 816	3 054	762	3 810
THU-FVFDT	610	8	4 880	3 904	976	6 832
FV-USM	492	6	2 952	2 364	588	2 940
UTFVP	360	4	1 440	1 152	288	864
VERA	220	2	440	352	88	88
SCUT-FV	568	6	3 408	2 730	678	3 390

$$average\_of\_metric = \frac{\sum_{i=1}^{9} NoP_i \times metric_i}{\sum_{i=1}^{9} NoP_i}. \quad (11)$$

4.3 Experiments on nine clients

In this section, the experiments compare the performance of FedFV with local training and the FedPav[33] algorithm to analyze the effect of federated learning on the performance of the local FV authentication system. The experimental results are shown in Table 2, with the best values in bold. From Table 2, we can see that the EER of each client has decreased under the FedPav and FedFV federated learning methods. Compared to local training, the EER of FedPav and FedFV is lower and the TAR@FAR = 0.01 is higher. The weighted average of the EERs of local training, FedPav, and FedFV are 3.38%, 1.64%, and 1.21%, respectively. These results demonstrate that FedPav and FedFV can effectively tackle the problem of FV data island and improve the FV authentication performance of the client's local model. Furthermore, it can show that FedPav and FedFV can learn the FV knowledge from other clients without touching their private data to increase the performance and robustness of its local model. The receiver operator characteristic (ROC) curves of FedFV on these nine FV datasets are shown in Fig. 2.

Furthermore, experiments show that the FedFV framework in this paper has lower EER for each client than FedPav, demonstrating that using FedFV provides local models with better FV feature extraction power. It can be found that all nine clients of FedFV have lower EER than FedPav, which demonstrates that FedFV can tackle the situation of strongly heterogeneous FV data. Experimental results show that the FedWPR algorithm used by FedFV can avoid excessive dilution of the client model in the smaller dataset and can learn from other clients while retaining personalized knowledge. Fig. 3 shows the bar chart of these experimental EERs and TAR@FAR = 0.01 for comparing the results of the local training and the proposed FedFV more intuitively.

4.4 Experiments of cooperating with different clients

The advantage of FedFV is that during the training process under this framework, clients can learn from other clients to improve the performance of their local models. Therefore, the dataset size of the collaborator will affect the performance improvement of the local model. In this work, a pairwise FedFV federated learning experiment is conducted on nine datasets to analyze the different collaborators in improving the performance of the model. A total of C_9^2 groups of experiments are carried out. The experimental results are shown in Table 3. The italic type denotes the worst performance cooperation, the bold font denotes the best cooperation, and the results in boxes denote the local training. From Table 3, we can see that, when cooperating with the remaining eight clients, client$_i$ will produce eight EERs, which are weighted average to get the Average of Cooperator metric. This metric can intuitively analyze the influence of adding collaborators on the performance of the local model. The calculation of the Average of Cooperator is shown in (12).

$$Average\,of\,Cooperator_i = \frac{\sum_{j=1}^{9} NoP_j \times EER_{ij}}{\sum_{j=1}^{9} NoP_j},\ j \neq i. \quad (12)$$

Table 3 shows that for the HKPU client, the EER of the local model is 2.8%, and the average of EER for cooperation is 1.77%. Meanwhile, the Averages of the Cooperator of eight clients are all smaller than the EER of local training. The experimental results in Table 3 demonstrate that cooperation with other clients can ba-

Table 2 Local training, newly federated learning, and FedFV on nine clients

FV datasets	Local train		FedPav		FedFV	
	EER (%)	TAR@FAR = 0.01	EER (%)	TAR@FAR = 0.01	EER (%)	TAR@FAR = 0.01
HKPU-FV	2.80	94.52	1.24	99.35	**0.48**	**99.57**
MMCBNU-6000	1.29	99.33	0.57	99.63	**0.35**	**99.85**
PLUSVein-FV	5.90	72.36	1.46	99.03	**0.76**	**99.31**
SDUMLA-HMT	3.96	89.29	2.07	97.22	**1.52**	**97.27**
THU-FVFDT	3.41	93.77	3.03	96.05	**2.09**	**97.00**
FV-USM	0.95	99.73	0.34	100.0	**0.07**	**100.0**
UTFVP	8.8	81.94	2.55	97.45	**1.62**	**99.57**
VERA	18.18	68.18	9.09	**86.36**	**4.54**	81.82
SCUT-FV	1.95	97.35	1.36	**99.65**	**0.82**	99.23
Average	2.64	94.72	1.49	98.48	**0.96**	**98.79**

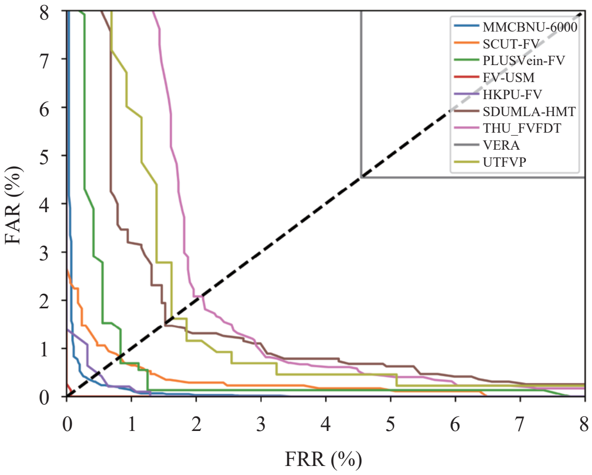


Fig. 2 ROC curve of FedFV on nine FV datasets. Curve of different FV datasets are draw by lines in different colors.

sically reduce EER compared to local training, indicating that federated learning can improve the performance of local models under the conditions of protecting private data.

In addition, we analyze how the improved performance of the local model differs by choosing different collaborators. As shown in Table 3, the performance of the local model is varied when cooperating with different clients. Collaborator A cooperates with eight other local models and will get eight EERs, which are averaged to obtain the Average of Client metric. The calculation of the Average of Clients is shown in (13), where NoP represents the number of authentication pairs, as shown in Table 1. Meanwhile, we average the EERs of eight local training except for collaborator A to obtain the Average of the Local metric, which is shown in (14). We next propose the Diff metric, which is shown in (15), to quantitatively describe the enhancement of model performance by selecting collaborator A.

$$Average\,of\,Client_j = \frac{\sum_{i=1}^{9} NoP_i \times EER_{ij}}{\sum_{i=1}^{9} NoP_{ij}},\ i \neq j \quad (13)$$

$$Average\,of\,Local_j = \frac{\sum_{i=1}^{9} NoP_i \times EER_{ii}}{\sum_{i=1}^{9} NoP_{ij}},\ i \neq j \quad (14)$$

$$Diff_j = Average\,of\,Local_j - Average\,of\,Client_j. \quad (15)$$

From Table 3, we can see that if the clients choose MMCBNU-6000, THU-FVFDT, or SCUT-FV as cooperators, the Diff is 1.36%, 1.21%, and 1.17%, respectively. However, if the clients decide on SDUMLA-HMT, UTFVP, or VERA as the cooperator, the Diff is 0.77%, 0.79%, and 0.92%, respectively. Combined with the results of Tables 1 and 3, it can be found that selecting collaborators with a large number of data such as MMCB-

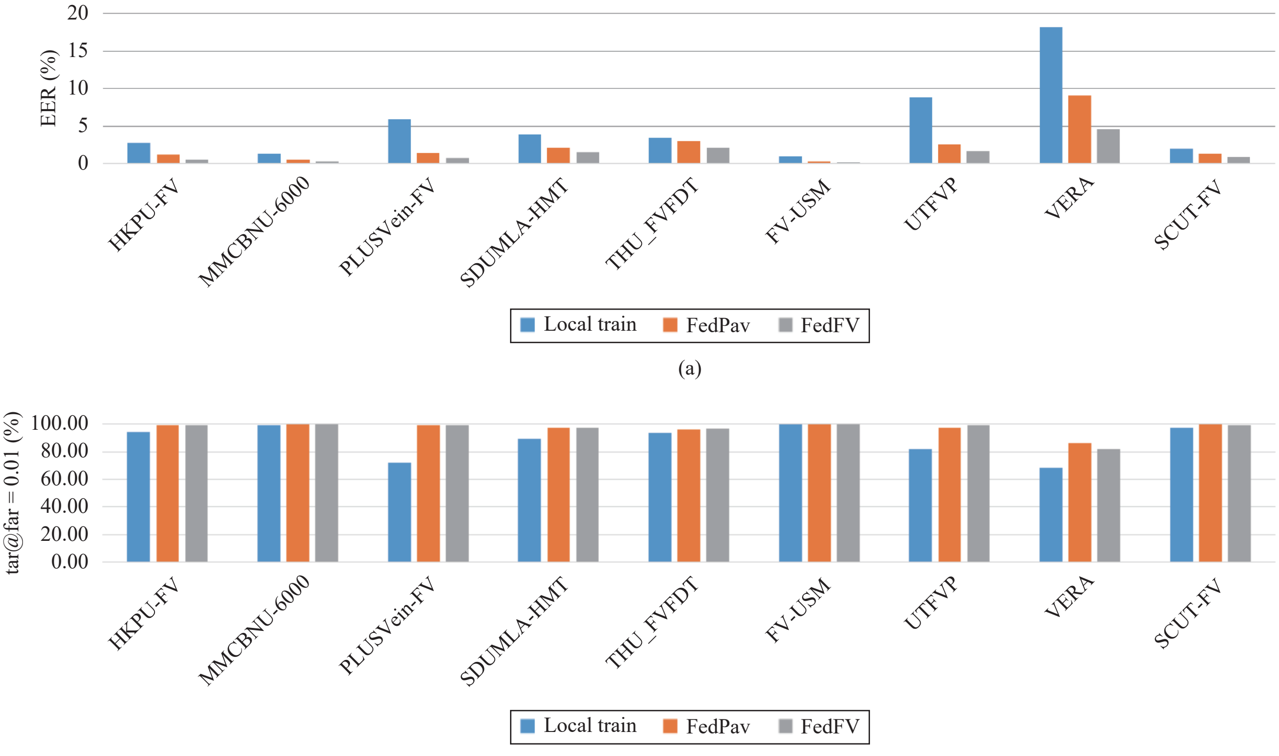


Fig. 3 Bar charts of the results of local training and federated learning in FV datasets. The bars in different colors denote the values of different models. (a) Bar charts of the EERs; (b) Bar charts of the TAR@FAR = 0.01.

Table 3 EERs (%) of cooperating with different clients

Cooperator(j)	Client(i)											
	HKPU-FV	MMCBNU-6000	PLUSVein-FV	SDUMLA-HMT	THU-FVFDT	FV-USM	UTFVP	VERA	SCUT-FV	Average of local	Average of client	Diff
HKPU-FV	$\boxed{2.8}$	0.82	2.36	2.94	2.39	0.41	5.79	11.36	1.21	2.63	1.69	0.94
MMCBNU-6000	1.94	$\boxed{1.29}$	1.81	2.34	2.78	**0.14**	3.12	**9.09**	1.12	3.33	1.97	**1.36**
PLUSVein-FV	1.99	0.79	$\boxed{5.9}$	2.57	2.49	0.31	4.75	13.64	1.3	2.49	1.62	0.87
SDUMLA-HMT	$\underline{2.2}$	**0.65**	3.06	$\boxed{3.96}$	2.99	0.34	4.86	13.64	$\underline{1.56}$	2.46	1.69	$\underline{0.77}$
THU-FVFDT	1.56	0.69	2.36	2.47	$\boxed{3.41}$	0.27	2.78	11.36	**1.01**	2.43	1.22	1.21
FV-USM	1.83	0.74	**1.74**	**2.13**	2.49	$\boxed{0.95}$	4.51	12.5	1.47	2.81	1.69	1.12
UTFVP	1.61	$\underline{1.05}$	3.26	2.31	2.56	$\underline{0.51}$	$\boxed{8.8}$	$\underline{15.91}$	1.21	2.47	1.68	0.79
VERA	2.04	0.66	3.33	2.7	2.44	0.48	$\underline{6.02}$	$\boxed{18.18}$	1.27	2.6	1.68	0.92
SCUT-FV	**1.08**	0.69	2.85	2.78	**2.37**	0.31	**2.78**	9.09	$\boxed{1.95}$	2.72	1.55	1.17
Average of cooperator	1.77	0.72	2.27	2.45	2.67	0.25	3.59	10.95	1.21	–	–	–

NU-6000, SCUT-FV, and THU-FVFDT for federated learning can significantly improve the FV authentication performance of the local model. While choosing collaborators with scarcity data such as UTFVP or VERA for federated learning, it will bring less improvement to the FV authentication performance of the local model. The experiment can demonstrate that in finger vein federated learning, the datasets with a large number of users can be preferred for partner selection.

4.5 Experiments under different number of clients

In this section, the experiments are conducted to analyze the performance influenced by the number of federated clients. Particularly, for each client, we incrementally increase the number of clients federated with it and record the performance for each federated. The experimental results are shown in Table 4. For clients HKPU-FV, FV-USM, VERA, etc., we can see that with the increase in the number of federate clients, the EER decreases gradually, and the model′s performance is steadily improved. However, for clients PLUSVein-FV, MMCBNU-6000, SCUT-FV, etc., we can see that when the number of federate clients increases, the EER does not always decline or is even larger. We analyze that the performance of each kind of client′s data has its upper limit of improvement and will not improve performance indefinitely with the increase of collaborators. These results imply that although federated learning can improve the performance of the model in general, however, the number of federated learning needs to be considered based on the specificity of the client. Fig. 4 shows the bar chart of these experimental EERs for comparing the results of the local training and the proposed FedFV more intuitively.

4.6 Comparison with the SOTA deep learning-based FV authentication methods

In this section, we compare the FedFV method with the SOTA of the existing FV authentication local training methods. Because some of the FV datasets are rarely

Table 4 EERs (%) of cooperating with different numbers of clients

Number of clients	HKPU-FV	MMCBNU-6000	PLUSVein-FV	SDUMLA-HMT	THU-FVFDT	FV-USM	UTFVP	SCUT-FV	VERA
1	2.8	1.29	5.9	3.96	3.41	0.95	8.8	1.98	18.18
2	1.99	0.82	2.36	2.94	2.99	0.41	5.79	1.47	11.36
3	1.67	0.69	2.29	2.6	2.88	0.37	4.98	1.12	9.09
4	1.56	0.59	1.81	2.6	2.63	0.37	3.7	1.12	9.09
5	1.18	0.52	1.67	2.39	2.53	0.34	2.89	1.21	9.09
6	1.08	0.63	1.25	2.26	2.59	0.27	2.66	1.06	9.09
7	0.81	0.46	1.39	2.02	2.44	0.27	2.66	1.06	9.09
8	0.75	0.49	1.3	2.2	2.59	0.2	2.31	1.06	6.82
9	**0.48**	**0.35**	**0.76**	**1.52**	**2.09**	**0.07**	**1.62**	**0.82**	**4.54**

Table 5 Comparison with the SOTA method

Paper	Method	EER (%)			
		MMCBNU-6000	SDUMLA-HMT	FV-USM	HKPU-FV
Hou and Yan[58]	Arccosine center loss	–	1.53	0.25	1.35
Yang et al.[21]	Joint recognition and anti-spoofing network	1.11	1.71	0.95	–
Hao et al.[18]	Multi-task learning(raw image)	**0.29**	**1.17**	0.74	–
Yang et al.[21]	VGG (reproduced by Yang et al.[21])	3.79	4.71	2.32	–
Yang et al.[21]	ResNet (reproduced by Yang et al.[21])	0.96	2.34	1.01	–
Yang et al.[13]	Finger vein code	–	–	–	3.33
Qin and El-Yacourbi[59]	Deep representation-based feature extraction	–	–	–	3.02
Huang et al.[22]	ViT (reproduced by Huang et al.[22])	1.74	5.77	1.63	5.48
Huang et al.[22]	Finger vein transformer (FVT)	0.92	1.5	0.44	2.37
Ours	FedFV	0.35	1.52	**0.07**	**0.48**

studied by local training methods, only the FV datasets that are often studied with local training are selected for comparison. The selected FV datasets are FV-USM, MMCBNU-6000, SDUMLA-HMT, and HKPU-FV. The results are shown in Table 5, and bold indicates the best value.

From Table 5, we can see that the performances of our proposed FedFV are superior to the current SOTA methods on the FV-USM and HKPU-FV. We can see that our proposed FedFV can learn the knowledge of other clients and improve the local model′s performance while protecting all clients′ data privacy. However, the EER of our proposed FedFV is still higher than the multi-task learning method in [18] on the MMCBNU-6 000 and SDUMLA-HMT. For this phenomenon, we analyze that the model in research[18] contains two branches and the input FV images are the raw images that can extract the FV information and the finger shape information. FedFV considers the communication cost and limits the number of parameters of the local model. Hence, the research[18] outperforms more than FedFV. Overall, under privacy protection and certain communication costs, FedFV can improve the performance of the local model through federated learning and tackle the problem of isolated FV data islands to some extent.

5 Conclusions

This work proposes a federated learning framework, FedFV, for alleviating the data deficiency problem of training the FV authentication system and solving the biometric template protection problem caused by cooperation between various clients. By using FedFV, the authentication models of each client can learn knowledge from all the federated participants and improve their performance without exposing their user data. Furthermore, to address the non-IID data problem caused by the client diversity during federated learning and enhance the personalization of each client′s model, this paper also proposes an efficient, personalized federated aggregation algorithm, FedWPR. The proposed FedFV framework can be effectively performed on different clients with different numbers of users, showing the universality of practical FV authentication scenarios. Extensive experiments on nine public FV datasets demonstrate the effectiveness of FedFV, which has some reference value for subsequent biometric privacy protection research.

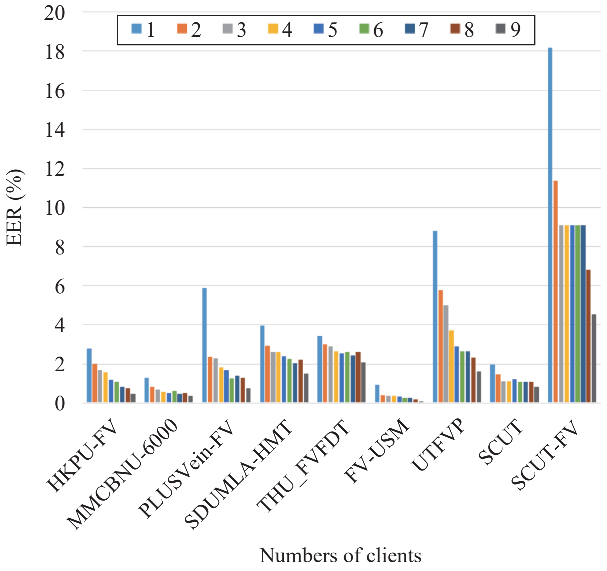


Fig. 4 Bar charts of the EERs cooperating with different numbers of clients. The bars in different colors denote the different numbers of clients.

Acknowledgements

This work was supported National Natural Science Foundation of China (No. 61976095) and Guangdong Province Science and Technology Planning Project, China (No. 2018B030323026).

References

[1] A. Uhl, C. Busch, S. Marcel, R. Veldhuis. *Handbook of Vascular Biometrics*, Cham, Switzerland: Springer, 2020. DOI: 10.1007/978-3-030-27731-4.

[2] L. Y. Xu, Z. Gajic. Improved network for face recognition based on feature super resolution method. *International Journal of Automation and Computing*, vol. 18, no. 6, pp. 915–925, 2021. DOI: 10.1007/s11633-021-1309-9.

[3] W. Jia, W. Xia, Y. Zhao, H. Min, Y. X. Chen. 2D and 3D palmprint and palm vein recognition based on neural architecture search. *International Journal of Automation and Computing*, vol. 18, no. 3, pp. 377–409, 2021. DOI: 10.1007/s11633-021-1292-1.

[4] W. Jia, J. Gao, W. Xia, Y. Zhao, H. Min, J. T. Lu. A performance evaluation of classic convolutional neural networks for 2D and 3D palmprint and palm vein recognition. *International Journal of Automation and Computing*, vol. 18, no. 1, pp. 18–44, 2021. DOI: 10.1007/s11633-020-1257-9.

[5] S. Tang, S. Zhou, W. X. Kang, Q. X. Wu, F. Q. Deng. Finger vein verification using a Siamese CNN. *IET Biometrics*, vol. 8, no. 5, pp. 306–315, 2019. DOI: 10.1049/iet-bmt.2018.5245.

[6] N. Miura, A. Nagasaka, T. Miyatake. Extraction of finger-vein patterns using maximum curvature points in image profiles. Extraction of finger-vein patterns using maximum curvature points in image profiles. *IEICE-TRANSACTIONS on Information and Systems*, vol. E90-D, no. 8, pp. 1185–1194, 2007. DOI: 10.1093/ietisy/e90-d.8.1185.

[7] W. Song, T. Kim, H. C. Kim, J. H. Choi, H. J. Kong, S. R. Lee. A finger-vein verification system using mean curvature. *Pattern Recognition Letters*, vol. 32, no. 11, pp. 1541–1547, 2011. DOI: 10.1016/j.patrec.2011.04.021.

[8] W. M. Yang, Z. Q. Chen, C. Qin, Q. M. Liao. α-trimmed weber representation and cross section asymmetrical coding for human identification using finger images. *IEEE Transactions on Information Forensics and Security*, vol. 14, no. 1, pp. 90–101, 2018. DOI: 10.1109/TIFS.2018.2844803.

[9] H. C. Lee, B. J. Kang, E. C. Lee, K. R. Park. Finger vein recognition using weighted local binary pattern code based on a support vector machine. *Journal of Zhejiang University SCIENCE C*, vol. 11, no. 7, pp. 514–524, 2010. DOI: 10.1631/jzus.C0910550.

[10] Y. T. Lu, M. Tu, H. Wang, J. H. Zhao, W. X. Kang. Finger vein recognition based on double-orientation coding histogram. In *Proceedings of the 14th Chinese Conference on Biometric Recognition*, Springer, Zhuzhou, China, pp. 20–27, 2019. DOI: 10.1007/978-3-030-31456-9_3.

[11] J. D. Wu, C. T. Liu. Finger-vein pattern identification using principal component analysis and the neural network technique. *Expert Systems with Applications*, vol. 38, no. 5, pp. 5423–5427, 2011. DOI: 10.1016/j.eswa.2010.10.013.

[12] F. Liu, Y. L. Yin, G. P. Yang, L. M. Dong, X. M. Xi. Finger vein recognition with superpixel-based features. In *Proceedings of IEEE International Joint Conference on Biometrics*, Clearwater, USA, pp. 1–8, 2014. DOI: 10.1109/BTAS.2014.6996232.

[13] L. Yang, G. P. Yang, X. M. Xi, K. Su, Q. Chen, Y. L. Yin. Finger vein code: From indexing to matching. *IEEE Transactions on Information Forensics and Security*, vol. 14, no. 5, pp. 1210–1223, 2019. DOI: 10.1109/TIFS.2018.2871778.

[14] W. X. Kang, Y. T. Lu, D. J. Li, W. Jia. From noise to feature: Exploiting intensity distribution as a novel soft biometric trait for finger vein recognition. *IEEE Transactions on Information Forensics and Security*, vol. 14, no. 4, pp. 858–869, 2019. DOI: 10.1109/TIFS.2018.2866330.

[15] Y. X. Fang, Q. X. Wu, W. X. Kang. A novel finger vein verification system based on two-stream convolutional network learning. *Neurocomputing*, vol. 290, pp. 100–107, 2018. DOI: 10.1016/j.neucom.2018.02.042.

[16] C. H. Xie, A. Kumar. Finger vein identification using Convolutional Neural Network and supervised discrete hashing. *Pattern Recognition Letters*, vol. 119, pp. 148–156, 2019. DOI: 10.1016/j.patrec.2017.12.001.

[17] H. C. Zheng, Y. J. Hu, B. B. Liu, G. Chen, A. C. Kot. A new efficient finger-vein verification based on lightweight neural network using multiple schemes. In *Proceedings of the 29th International Conference on Artificial Neural Networks*, Springer, Bratislava, Slovakia, pp. 748–758, 2020. DOI: 10.1007/978-3-030-61609-0_59.

[18] Z. A. Hao, P. Y. Fang, H. W. Yang. Finger vein recognition based on multi-task learning. In *Proceedings of the 5th International Conference on Mathematics and Artificial Intelligence*, ACM, Chengdu, China, pp. 133–140, 2020. DOI: 10.1145/3395260.3395277.

[19] J. D. Huang, M. Tu, W. L. Yang, W. X. Kang. Joint attention network for finger vein authentication. *IEEE Transactions on Instrumentation and Measurement*, vol. 70, Article number 2513911, 2021. DOI: 10.1109/TIM.2021.3109978.

[20] R. S. Kuzu, E. Piciucco, E. Maiorana, P. Campisi. On-the-fly finger-vein-based biometric recognition using deep neural networks. *IEEE Transactions on information Forensics and Security*, vol. 15, pp. 2641–2654, 2020. DOI: 10.1109/TIFS.2020.2971144.

[21] W. L. Yang, W. Luo, W. X. Kang, Z. X. Huang, Q. X. Wu. FVRAS-Net: An embedded finger-vein recognition and antiSpoofing system using a unified CNN. *IEEE Transactions on Instrumentation and Measurement*, vol. 69, no. 11, pp. 8690–8701, 2020. DOI: 10.1109/TIM.2020.3001410.

[22] J. D. Huang, W. J. Luo, W. L. Yang, A. Zheng, F. Z. Lian, W. X. Kang. FVT: Finger vein transformer for authentication. *IEEE Transactions on Instrumentation and Measurement*, vol. 71, Article number 5011813, 2022. DOI: 10.1109/TIM.2022.3173276.

[23] B. McMahan, E. Moore, D. Ramage, S. Hampson, B. A. Y. Arcas. Communication-efficient learning of deep networks from decentralized data. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, Fort Lauderdale, USA, pp. 1273–1282, 2017.

[24] S. Ramaswamy, R. Mathews, K. Rao, F. Beaufays. Federated learning for emoji prediction in a mobile keyboard. [Online], Available: https://arxiv.org/abs/1906.04329, 2019.

[25] T. Yang, G. Andrew, H. Eichner, H. C. Sun, W. Li, N. Kong, D. Ramage, F. Beaufays. Applied federated learning: Improving Google keyboard query suggestions. [Online], Available: https://arxiv.org/abs/1812.02903, 2018.

[26] Y. Zhao, M. Li, L. Z. Lai, N. Suda, D. Civin, V. Chandra. Federated learning with non-ⅡD data. [Online], Available: https://arxiv.org/abs/1806.00582, 2018.

[27] K. Hsieh, A. Phanishayee, O. Mutlu, P. Gibbons. The non-ⅡD data quagmire of decentralized machine learning. In *Proceedings of the 37th International Conference on Machine Learning*, pp. 4387–4398, 2020.

[28] A. K. Sahu, T. Li, M. Sanjabi, M. Zaheer, A. Talwalkar, V. Smith. On the convergence of federated optimization in heterogeneous networks. [Online], Available: https://arxiv.org/abs/1812.06127v1, 2018.

[29] F. Bai, J. X. Wu, P. C. Shen, S. X. Li, S. G. Zhou. Federated face recognition. [Online], Available: https://arxiv.org/abs/2105.02501, 2021.

[30] Z. H. Tang, Z. K. Hu, S. H. Shi, Y. M. Cheung, Y. L. Jin, Z. H. Ren, X. W. Chu. Data resampling for federated learning with non-ⅡD labels. In *Proceedings of the International Workshop on Federated and Transfer Learning for Data Sparsity and Confidentiality in Conjunction with IJCAI 2021*, 2021.

[31] A. Z. Tan, H. Yu, L. Z. Cui, Q. Yang. Towards personalized federated learning. *IEEE Transactions on Neural Networks and Learning Systems*, to be published. DOI: 10.1109/TNNLS.2022.3160699.

[32] Y. Q. Chen, X. Qin, J. D. Wang, C. H. Yu, W. Gao. FedHealth: A federated transfer learning framework for wearable healthcare. *IEEE Intelligent Systems*, vol. 35, no. 4, pp. 83–93, 2020. DOI: 10.1109/MIS.2020.2988604.

[33] W. M. Zhuang, Y. G. Wen, X. S. Zhang, X. Gan, D. Y. Yin, D. Z. Zhou, S. Zhang, S. Yi. Performance optimization of federated person re-identification via benchmark analysis. In *Proceedings of the 28th ACM International Conference on Multimedia*, ACM, Seattle, USA, pp. 955–963, 2020. DOI: 10.1145/3394171.3413814.

[34] S. Itahara, T. Nishio, Y. Koda, M. Morikura, K. Yamamoto. Distillation-based semi-supervised federated learning for communication-efficient collaborative training with non-ⅡD private data. *IEEE Transactions on Mobile Computing*, to be published. DOI: 10.1109/TMC.2021.3070013.

[35] V. Smith, C. K. Chiang, M. Sanjabi, A. Talwalkar. Federated multi-task learning. *Proceedings of the 31st Conference on Neural Information Processing Systems*, Long Beach, USA, pp. 4424–4434, 2017.

[36] T. Li, S. Y. Hu, A. Beirami, V. Smith. Ditto: Fair and robust federated learning through personalization. In *Proceedings of the 38th International Conference on Machine Learning*, pp. 6357–6368, 2021.

[37] T. Li, S. Y. Hu, A. Beirami, V. Smith. Ditto: Fair and robust federated learning through personalization. [Online], Available: https://arxiv.org/abs/2012.04221, 2020.

[38] J. Li, M. Khodak, S. Caldas, A. Talwalkar. Differentially private meta-learning. [Online], Available: https://arxiv.org/abs/1909.05830, 2019.

[39] F. Chen, M. Luo, Z. H. Dong, Z. G. Li, X. Q. He. Federated meta-learning with fast convergence and efficient communication. [Online], Available: https://arxiv.org/abs/1802.07876, 2018.

[40] A. Fallah, A. Mokhtari, A. Ozdaglar. Personalized federated learning: A meta-learning approach. [Online], Available: https://arxiv.org/abs/2002.07948, 2020.

[41] Y. H. Jiang, J. Konečnỳ, K. Rush, S. Kannan. Improving federated learning personalization via model agnostic meta learning. [Online], Available: https://arxiv.org/abs/1909.12488, 2019.

[42] F. Sattler, K. R. Muller, W. Samek. Clustered federated learning: Model-agnostic distributed multitask optimization under privacy constraints. *IEEE Transactions on Neural Networks and Learning Systems*, vol. 32, no. 8, pp. 3710–3722, 2021. DOI: 10.1109/TNNLS.2020.3015958.

[43] L. Huang, A. L. Shea, H. N. Qian, A. Masurkar, H. Deng, D. B. Liu. Patient clustering improves efficiency of federated machine learning to predict mortality and hospital stay time using distributed electronic medical records. *Journal of Biomedical Informatics*, vol. 99, Article number 103291, 2019. DOI: 10.1016/j.jbi.2019.103291.

[44] Y. T. Huang, L. Y. Chu, Z. R. Zhou, L. J. Wang, J. C. Liu, J. Pei, Y. Zhang. Personalized cross-silo federated learning on non-ⅡD data. In *Proceedings of the 35th AAAI Conference on Artificial Intelligence*, pp. 7865–7873, 2021.

[45] Z. Chai, Y. J. Chen, L. Zhao, Y. Cheng, H. Rangwala. FedAT: A communication-efficient federated learning method with asynchronous tiers under non-ⅡD data. [Online], Available: https://arxiv.org/abs/2010.05958, 2020.

[46] l. Masi, Y. Wu, T. Hassner, P. Natarajan. Deep face recognition: A survey. In *Proceedings of the 31st SIBGRAPI Conference on Graphics, Patterns and Images*, IEEE, Parana, Brazil, pp. 471478, 2018. DOI: 10.1109/SIBGRAPI.2018.00067.

[47] M. Sandler, A. Howard, M. L. Zhu, A. Zhmoginov, L. C. Chen. MobileNetV2: Inverted residuals and linear bottlenecks. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition*, IEEE, Salt Lake City, USA, pp. 4510–4520, 2018. DOI: 10.1109/CVPR.2018.00474.

[48] P. T. de Boer, D. P. Kroese, S. Mannor, R. Y. Rubinstein. A tutorial on the cross-entropy method. *Annals of Operations Research*, vol. 134, no. 1, pp. 19–67, 2005. DOI: 10.1007/s10479-005-5724-z.

[49] Y. D. Wen, K. P. Zhang, Z. F. Li, Y. Qiao. A discriminative feature learning approach for deep face recognition. In *Proceedings of the 14th European Conference on Computer Vision*, Springer, Amsterdam, The Netherlands, pp. 499–515, 2016. DOI: 10.1007/978-3-319-46478-7_31.

[50] Y. Lu, S. J. Xie, S. Yoon, Z. H. Wang, D. S. Park. An available database for the research of finger vein recogni-

tion. In *Proceedings of the 6th International Congress on Image and Signal Processing*, IEEE, Hangzhou, China, pp. 410–415, 2013. DOI: 10.1109/CISP.2013.6744030.

[51] A. Kumar, Y. B. Zhou. Human identification using finger images. *IEEE Transactions on Image Processing*, vol. 21, no. 4, pp. 2228–2244, 2012. DOI: 10.1109/TIP.2011.2171697.

[52] C. Kauba, B. Prommegger, A. Uhl. Focussing the beam - A new laser illumination based data set providing insights to finger-vein recognition. In *Proceedings of the 9th International Conference on Biometrics Theory, Applications and Systems*, IEEE, Redondo Beach, USA, pp. 1–9, 2018. DOI: 10.1109/BTAS.2018.8698588.

[53] Y. L. Yin, L. L. Liu, X. W. Sun. SDUMLA-HMT: A multimodal biometric database. In *Proceedings of the 6th Chinese Conference on Biometric Recognition*, Springer, Beijing, China, pp. 260–268, 2011. DOI: 10.1007/978-3-642-25449-9_33.

[54] W. M. Yang, C. Qin, Q. M. Liao. A database with ROI extraction for studying fusion of finger vein and finger dorsal texture. In *Proceedings of the 9th Chinese Conference on Biometric Recognition*, Springer, Shenyang, China, pp. 266–270, 2014. DOI: 10.1007/978-3-319-12484-1_30.

[55] M. S. M. Asaari, S. A. Suandi, B. A. Rosdi. Fusion of band limited phase only correlation and width centroid contour distance for finger based biometrics. *Expert Systems with Applications*, vol. 41, no. 7, pp. 3367–3382, 2014. DOI: 10.1016/j.eswa.2013.11.033.

[56] B. T. Ton, R. N. J. Veldhuis. A high quality finger vascular pattern dataset collected using a custom designed capturing device. In *Proceedings of International Conference on Biometrics*, IEEE, Madrid, Spain, 2013. DOI: 10.1109/ICB.2013.6612966.

[57] P. Tome, M. Vanoni, S. Marcel. On the vulnerability of finger vein recognition to spoofing. In *Proceedings of International Conference of the Biometrics Special Interest Group*, IEEE, Darmstadt, Germany, pp. 111–120, 2014.

[58] B. R. Hou, R. Q. Yan. ArcVein-arccosine center loss for finger vein verification. *IEEE Transactions on Instrumentation and Measurement*, vol. 70, Article number 5007411, 2021. DOI: 10.1109/TIM.2021.3062164.

[59] H. F. Qin, M. A. El-Yacoubi. Deep representation-based feature extraction and recovering for finger-vein verification. *IEEE Transactions on Information Forensics and Security*, vol. 12, no. 8, pp. 1816–1829, 2017. DOI: 10.1109/TIFS.2017.2689724.

Feng-Zhao Lian received the B. Sc. degree in automation from South China University of Technology, China in 2019. He is currently a master student at School of Automation Science and Engineering, South China University of Technology, China.

His research interests include biometrics, federated learning, computer vision and deep learning.

E-mail: lianfengzhaoscut@qq.com

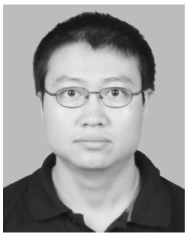
ORCID iD: 0000-0002-4219-067X

Jun-Duan Huang received the B. Sc. degree in automation, and M. Sc. degree in agricultural electrification and automation, from South China Agriculture University, Guangzhou, China, in 2017 and 2020, respectively. He is currently a doctoral candidate in electronic and information at South China University of Technology, Guangzhou, China.

His research interests include biometrics, computer vision, audio signal processing, deep learning and agricultural engineering.

E-mail: runrunjun@163.com

ORCID iD: 0000-0002-5510-7046



Ji-Xin Liu received the M. Sc. degree in power electronics and power drives from Northeast Petroleum University, China in 2004, and the Ph. D. degree in information and communication engineering from Harbin Institute of Technology, China in 2010. He is currently an associate professor with School of Automation, Guangdong University of Petrochemical Technology, China.

His research interests include biometric identification, privacy preserving machine learning, pattern recognition and fault diagnosis of petrochemical equipment.

E-mail: ljxfrog@qq.com

Guang Chen received the B. Sc. degree in mechatronics engineering from Xi'an University of Architecture and Technology, China in 2008.

His research interests include biometric recognition, machine learning and federated learning.

E-mail: cguang1@grgbanking.com

Jun-Hong Zhao received the M. Sc. degree in pattern recognition and intelligent system from Chongqing University, China in 2003, and the Ph. D. degree in pattern recognition and intelligent system from South China University of Technology, China in 2011. She is currently a lecturer with School of Automation Science and Engineering, South China University of Technology, China.

Her research interests include image processing, image forensics and biometrics identification.

E-mail: jhzhao@scut.edu.cn (Corresponding author)

Wen-Xiong Kang received the Ph. D. degree in systems engineering from South China University of Technology, China in 2009. He is currently a professor with School of Automation Science and Engineering, South China University of Technology, China.

His research interests include biometrics identification, image processing, pattern recognition, and computer vision.

E-mail: auwxkang@scut.edu.cn (Corresponding author)