Federated Learning for Face Recognition via Intra-subject Self-supervised Learning

Hansol Kim[1,2]
hans.kim@kakaobank.com
Hoyeol Choi[1]
con.choi@kakaobank.com
Youngjun Kwak[†,1,3]
vivaan.yjkwak@kakaobank.com,yjk.kwak@kaist.ac.kr

[1] KakaoBank Corp., South Korea
[2] Graduate School of Software Technology, Kookmin University, South Korea
[3] Department of Electrical Engineering, KAIST, South Korea

Abstract

Federated Learning (FL) for face recognition aggregates locally optimized models from individual clients to construct a generalized face recognition model. However, previous studies present two major challenges: insufficient incorporation of self-supervised learning and the necessity for clients to accommodate multiple subjects. To tackle these limitations, we propose FedFS (Federated Learning for personalized Face recognition via intra-subject Self-supervised learning framework), a novel federated learning architecture tailored to train personalized face recognition models without imposing subjects. Our proposed FedFS comprises two crucial components that leverage aggregated features of the local and global models to cooperate with representations of an off-the-shelf model. These components are (1) adaptive soft label construction, utilizing dot product operations to reformat labels within intra-instances, and (2) intra-subject self-supervised learning, employing cosine similarity operations to strengthen robust intra-subject representations. Additionally, we introduce a regularization loss to prevent overfitting and ensure the stability of the optimized model. To assess the effectiveness of FedFS, we conduct comprehensive experiments on the DigiFace-1M and VGGFace datasets, demonstrating superior performance compared to previous methods.

1 Introduction

Recent years have witnessed a burgeoning interest in safeguarding personal data, a concern further emphasized by Article 25 of the GDPR [26], which mandates heightened data protection measures throughout system development and prohibits the unauthorized collection of personal information. Consequently, safeguarding personal information during the training of deep-learning networks has emerged as a paramount concern.

Face recognition has garnered considerable attention due to its efficacy in identifying individuals. This method finds widespread application in user authentication and has even found integration into smartphones to safeguard personal information or financial transactions. However, a significant portion of face recognition models [14, 15, 19] are typically

† Corresponding author

hosted on servers, necessitating the transmission of facial images from smartphones for authentication, which raises privacy concerns. To address this issue, the adoption of lightweight models directly on smartphones has been proposed. Nonetheless, limitations persist in training these models solely using public data. Consequently, there is a growing interest in training models to utilize users' facial data on their own devices while ensuring data privacy, with many studies leveraging federated learning methods garnering attention in this regard.

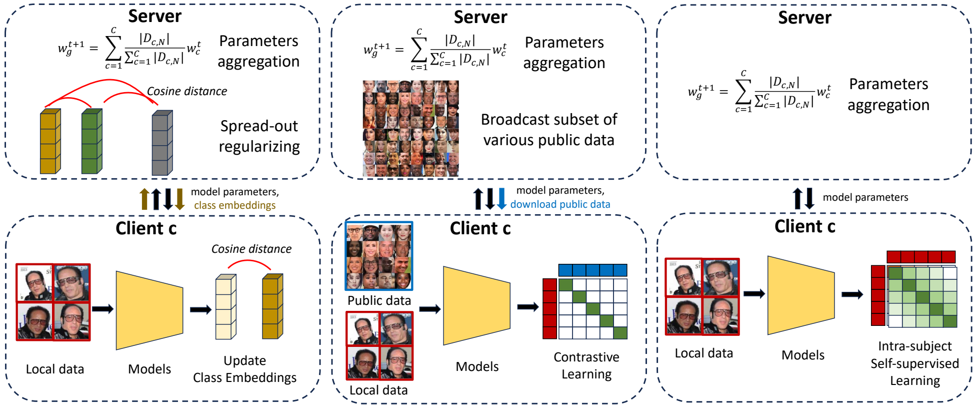
Federated learning is a method in which multiple clients join together to train a model with good performance while protecting personal information. FedFace [1] introduced a spread-out regularizer aimed at training a face recognition model within a federated learning framework. However, the process of dispersing the identity proxies received from clients in FedFace raises concerns regarding potential privacy violations. FedFR [20] prevented bias by training personalized models using public data, demonstrating promising performance among federated learning-based face recognition models. However, this approach necessitates clients to continuously receive public data, posing significant resource constraints, especially in on-device environments like mobile platforms where computational resources are severely limited. Additionally, FedFR proposed a new evaluation metric for personalizing performance, but this metric is far from real-world situations because the number of clients is too small and one client holds multiple identities.

To address these challenges, we propose FedFS, which trains generalized facial features and personalized face recognition model without leaking personal data outside each user's device in a federated learning environment. FedFS has three models: a pre-trained model trained with public data, a personalized model, and a global model, as well as two components: adaptive soft label construction utilizing dot product and intra-subject self-supervised learning employing cosine similarity, to reduce computational complexity and intra-class variation. Additionally, we introduce regularization loss to prevent bias in personalized models in heterogeneous data situations. We assume an actual authentication environment in which tens of thousands of clients participate in federated learning and evaluate personalizing performance using DigiFace-1M [3] and VGGface [6] benchmark data. Our main contributions are summarized as follows:

- We propose FedFS, Federated Learning for personalized Face recognition via intra-subject Self-supervised learning framework. FedFS trains optimized facial features for each client and reduces intra-class variation by leveraging adaptive soft label construction utilizing dot product and intra-subject self-supervised learning employing cosine similarity while protecting users' data privacy.

- Regularization loss is proposed to prevent bias in the performance of personalized models. Through this, FedFS solves the problem of easily falling into overfitting when training only with personal data, and trains indirectly generalized facial features.

- Experiment results utilizing face recognition benchmarks like DigiFace-1M and VGGFace demonstrate that our proposed method outperforms previous approaches. Furthermore, we conducted training and evaluation with the assumption of 10,000 clients participating, each with only one identity, mirroring real-world conditions. This assumption is the first attempt in federated learning-based face recognition research.


Server
$w_g^{t+1} = \sum_{c=1}^{C} \frac{|D_{c,N}|}{\sum_{c=1}^{C} |D_{c,N}|} w_c^t$
Parameters aggregation
Cosine distance
Spread-out regularizing
model parameters, class embeddings
Client c
Cosine distance
Local data
Models
Update Class Embeddings
Server
$w_g^{t+1} = \sum_{c=1}^{C} \frac{|D_{c,N}|}{\sum_{c=1}^{C} |D_{c,N}|} w_c^t$
Parameters aggregation
Broadcast subset of various public data
model parameters, download public data
Client c
Public data
Local data
Models
Contrastive Learning
Server
$w_g^{t+1} = \sum_{c=1}^{C} \frac{|D_{c,N}|}{\sum_{c=1}^{C} |D_{c,N}|} w_c^t$
Parameters aggregation
model parameters
Client c
Local data
Models
Intra-subject Self-supervised Learning

(a) Share class embeddings and parameters (b) Download public data and share parameters (c) Share only parameters (Ours)

Figure 1: Pipelines of federated learning-based face recognition methods including our proposed method. (a) The server collects class embedding of client c (e.g. FedFace). (b) Client c continuously downloads public data from the server (e.g. FedFR). (c) Our proposed method(FedFS), client c performs intra-subject self-supervised learning without any additional work.

2 Related Works

Face Recognition. Face recognition has seen a remarkable enhancement in performance through the utilization of large-scale data, identities, and models, sparking considerable interest [4, 17, 28]. However, state-of-the-art models require a lot of resources, so the execution environment is often limited to servers with no resource limitations. In this case, personal information is violated because the actual authentication process requires facial data to be transmitted from the client to the server. In contrast to large-scale models, there is a growing body of research focusing on lightweight models [2, 5]. MobileFaceNet [2] exemplifies one such lightweight face recognition model, boasting speeds that are more than twice as fast as MobileNetV2 [23]. However, enhancing the performance of pre-trained models like these, which utilize public data, proves challenging due to constraints imposed by model size and the inability to conduct additional training using user data. FedFS aims to address this limitation by enhancing recognition performance through personalized facial feature training while safeguarding personal information.

Federated Learning. Federated learning is attracting attention as a way to protect personal information. FedAvg [22] is a method of creating a global model by calculating the average of the parameters of a local model trained using data from each client. Recently, research on personalized federated learning, which improves personalized performance by utilizing models customized to suit individual goals, is increasing. However, most research has only been conducted on small-scale datasets such as MNIST [18] and CIFAR-10 [16]. To solve these problems, research on face recognition has been conducted in federated learning environments such as FedFace [1] and FedFR [20]. FedFR simultaneously trained for generalizing performance and personalizing performance using public data. In contrast, we do not use public data directly, because utilizing the data requires the client's resources, which can be very taxing on the client's devices.

Contrastive Learning. Contrastive learning researchs achieve state-of-the-art results on learning image features [8, 9, 12]. The main idea of contrastive learning is to diminish the distance between the features of the same identity of the images and increase the distance between the features of different identities of images. In the past, dot products were widely used in contrastive learning, but recently, cosine similarity has been widely used [10, 27]. This shift is attributed to the potential of dot products to yield large values based on the data, resulting in various issues such as inflated weight values [25]. However, geometrically speaking, cosine similarity solely concerns angular, rendering normalized data indistinguishable in magnitude. This means that cosine similarity is effective in maximizing inter-class variation, but shows poor performance in some cases [24]. In contrast, the dot product is influenced not only by the angle but also by the magnitude, enabling differentiation even among data involving the same identifier. From this perspective, we aim to minimize intra-class variation by using dot product and cosine similarity simultaneously.

Contrastive learning is garnering significant attention due to its outstanding performance, and numerous studies applying federated learning and contrastive learning are underway. Unlike traditional contrastive learning approaches, in federated contrastive learning, clients can only have their data, so there are no other identities. To address this challenge, a variety of federated learning-based studies [13, 20] are attracting much attention. In this paper, we focus on effectively learning individual features (positive data) without other identities (negative data) in a federated learning setup and propose regularization loss to prevent overfitting and bias.

Algorithm 1 Procedure of FedFS

```
Communication Round is $t, t \in \{0,...,T\}$.
Initialize a server-global model parameters $w_g^0$.
Broadcast pre-trained model $\xi$ to all participating clients.
Server executes:
for $t = 0,...,T$ do
    for $c = 1,...,C$ do
        Send the server-global model parameters $w_g^t$ to client c
        $w_c^t$ and $|D_{c,N}| \leftarrow$ ClientTraining($w_g^t$)
    end for
    $w_g^{t+1} = \sum_{c=1}^{C} \frac{|D_{c,N}|}{\sum_{c=1}^{C} |D_{c,N}|} w_c^t$
end for

function ClientTraining($w_g^t$):
$w_c^t \leftarrow w_g^t$
for $i = 1,...,N$ do
    $F_{total}(\psi, w_c, \theta_c) = \lambda * F_{insub}(\psi, w_c, \theta_c) + (1-\lambda) * F_{reg}(w_c, \theta_c)$
    $w_c^t \leftarrow w_c^t - \eta \nabla_{w_c} F_{total}(\psi, w_c, \theta_c)$
    $\theta_c^t \leftarrow \theta_c^t - \eta \nabla_{\theta_c} F_{total}(\psi, w_c, \theta_c)$
end for
Calculate the number of the c client data $D_{c,N}$
return $w_c^t$ and $|D_{c,N}|$
end function
```

3 Proposed Method

In this section, we propose a federated learning framework for personalized face recognition with intra-subject self-supervised learning and this flow is summarized in Algorithm 1. We will first describe the training environment and then explain in detail the training process proposed in this environment. Additionally, we demonstrate the convergence analysis for our proposed method in Appendix **??**.

3.1 Problem Formulation

We define the total number of participating clients in federated learning as C, and the specific client as c, $c \in \{1,...,C\}$. Clients combine a personalized model, a pre-trained model, and a global model to collectively train their individual facial features. The personalized model has the same architecture as the global model. Each client has a training dataset $D_{C,N} = \{x_{c,i}, 1 \leq i \leq N, 1 \leq c \leq C\}$, where N is the cardinality of the local data $D_{C,N}$. In a federated learning setup, the parameter server collects and aggregates the parameters of the global model from each client without sharing any private data. We adopt the commonly used FedAvg [22] as our aggregate baseline method. This step is summarized as follows:

$$w_g^{t+1} = \sum_{c=1}^{C} \frac{|D_{c,N}|}{\sum_{c=1}^{C} |D_{c,N}|} w_c^t \tag{1}$$

where t means tth communication round, $t \in \{0,...,T\}$, w_g is the parameters of server-global model, w_c is the parameters of global model trained in personal device of client c and $|D_{c,N}|$ is the number of samples on dataset $D_{c,N}$. After updating the parameters of the server-global model, the parameters are broadcast to all clients. Through this process, we can indirectly train generalized facial features.

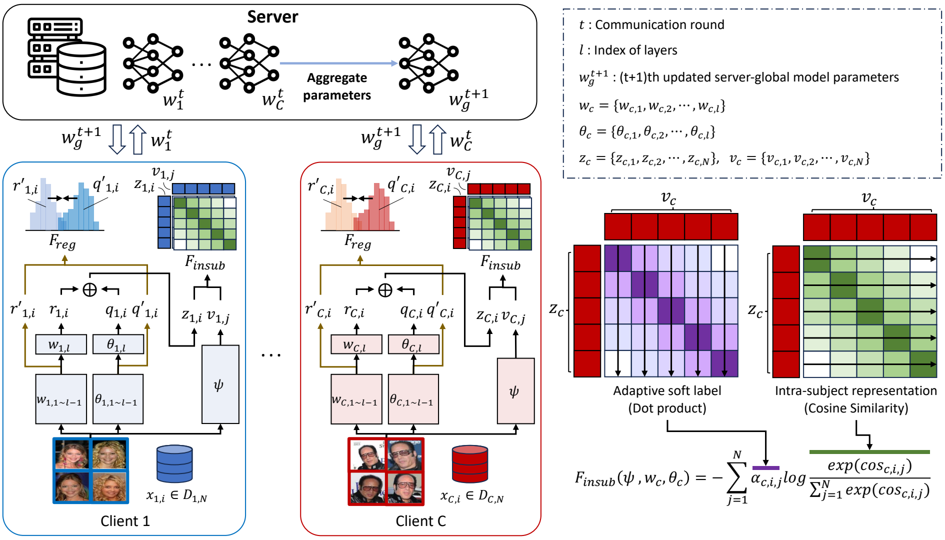
3.2 Intra-subject self-supervised learning

Intra-subject representations. In intra-subject self-supervised learning, two major operations are performed simultaneously. 1) Training local information and reducing intra-class variation with intra-subject loss. 2) Preventing overfitting and bias with regularization loss. Considering the client's restriction to utilize only local data for privacy protection, each client trains the model using only positive data, excluding negative data. Under these conditions, the client c performs operations with the global model, personalized model, and pre-trained model on input data $x_{c,i}$ and obtains the following results, respectively:

$$r_{c,i} = \phi_c(x_{c,i}, w_c), \quad q_{c,i} = \phi_c(x_{c,i}, \theta_c), \quad v_{c,i} = \xi(x_{c,i}, \psi), \quad z_{c,i} = r_{c,i} \oplus q_{c,i} \tag{2}$$

where w_c is the global model($\phi(w)$) parameters of client c, θ_c is the personalized model($\phi(\theta)$) parameters of client c and ψ is the pre-trained model(ξ) parameters. Subsequently, we obtain the intra-subject representation using the cosine similarity between the results and calculate the intra-subject loss value within the online-batch. We do not update ψ parameters, and share θ_c parameters with the server. The process is as follows:

$$cos_{c,i,j} = 1 - \frac{z_{c,i} \cdot v_{c,j}}{||z_{c,i}||_2 \cdot ||v_{c,j}||_2} \tag{3}$$



(a) Intra-subject Self-supervised Learning

(b) Intra-subject Loss

Figure 2: (a) is an overview of our proposed training process and (b) is the detailed process of intra-subject loss. The global model outputs two vectors and the personalized model also outputs two vectors. Using each output, we calculate regularization loss and create a z_c vector. Intra-subject loss is measured using the z_c vector and the output vector of the pre-trained model.

$$F_{insub}(\psi, w_c, \theta_c) = -\sum_{j=1}^{N} y_{c,j} log \frac{exp(cos_{c,i,j})}{\sum_{j=1}^{N} exp(cos_{c,i,j})} \tag{4}$$

where $y_{c,j}$ means the class label, and j has the same meaning as i $\{x_{c,j}, 1 \leq j \leq N\}$, but used to distinguish v_c and z_c. The data $x_{c,i-1}$, $x_{c,i}$, and $x_{c,i+1}$ whthin the online-batch are all positive data, so each data have a high similarity to each other. However, due to the nature of cross entropy, $y_{c,j}$ is 0 except in cases where the input is the exactly same image within the online-batch. To address these limitations, our proposed method uses an adaptive soft label that reflects the correlation between all positive data to reformat labels within intra-instances and reduce intra-class variance, thereby more effectively training correlations for local data.

Adaptive soft label. To obtain an adaptive soft label, we calculate the adaptive soft score *ass* using the dot product. Additionally, we select the K ratio of the batch size in descending order to emphasize the correlation with the specific ratio. Afterward, instead of labels, we use the adaptive soft label. This process is as follows:

$$ass_{c,i,j} = z_{c,i} \cdot v_{c,j}, \quad ASS_{c,i,j} \in \{ass_{c,1,j}, ..., ass_{c,i,j}\} \tag{5}$$

$$\beta_{c,i,j} \begin{cases} ass_{c,i,j} * \gamma, & \text{if } y_{c,j} = 1 \\ ass_{c,i,j}, & \text{else if } \star \\ 0, & \text{otherwise} \end{cases}$$

$$\alpha_{c,i,j} = (\frac{exp(\beta_{c,i,j})}{\sum_{i=1}^{N} exp(\beta_{c,i,j})})^T \tag{6}$$

where $\star$ means $y_{c,j} = 0$ and $ass_{c,i,j}$ is Top K in $ASS_{c,i,j}$. $ASS_{c,i,j}$ is adaptive soft score set of client c. The γ is a positive number, and the K value is a value between 0 and 1. The γ is a probability enhancement value for itself. As the value increases, the model focuses more on training the similarity to itself, and as it becomes smaller, the model trains by focusing on the similarity to surrounding vectors. In this paper, the K value is set to 4 and γ is set to 2. Finally, the local facial features are trained by performing a cross-entropy operation using the adaptive soft label α instead of the previously used label value $y_{c,j}$. The process can be summarized as follows:

$$F_{insub}(\psi, w_c, \theta_c) = -\sum_{j=1}^{N} \alpha_{c,i,j} log \frac{exp(cos_{c,i,j})}{\sum_{j=1}^{N} exp(cos_{c,i,j})} \tag{7}$$

Regularization loss. Training only on local data without including negative data can easily lead to overfitting and biased results. To solve this problem, we perform regularizing between the global model that trains generalized facial features through sharing the parameters with the server and the personalized model, as follows:

$$F_{reg}(w_c, \theta_c) = 1 - \frac{r'_{c,i} \cdot q'_{c,i}}{||r'_{c,i}||_2 \cdot ||q'_{c,i}||_2} \tag{8}$$

where $r'_{c,i}$ and $q'_{c,i}$ are the output vectors that do not pass through the last linear layer of the global model and personalized model, respectively. Finally, the intra-subject self-supervised learning process is summarized as follows.

$$F_{total}(\psi, w_c, \theta_c) = \lambda * F_{insub}(\psi, w_c, \theta_c) + (1 - \lambda) * F_{reg}(w_c, \theta_c) \tag{9}$$

where λ is an objective weight value between 0 and 1. In this paper, λ is set to 0.7.

4 Experiments

In this section, we demonstrate the performance of our proposed method through experiments. To evaluate the performance of each client's personalized face recognition model, we use an evaluation technique that arranges the evaluation data in a 1:N structure conducted in FedFR [20]. In addition, we check whether our proposed method reduces intra-class variation and ablation study in Appendix **??**.

Pre-trained model	FL method	DigiFace-1M		VGGFace	
		AUROC	%	AUROC	%
MobileFaceNet	-	0.8248	-	0.8921	-
	FedFace	0.5001	-60.6%	0.5488	-61.51%
	FedFR	0.8270	+0.2%	0.9477	+6.2%
	FedFS(Ours)	**0.9629**	**+16.7%**	**0.9794**	**+9.7%**
GhostFaceNets	-	0.9612	-	0.9885	-
	FedFace	0.5106	-53.1%	0.5905	-59.7%
	FedFR	0.9644	+0.3%	0.9929	+0.4%
	FedFS(Ours)	**0.9944**	**+3.4%**	**0.9943**	**+0.5%**
PocketNet	-	0.9128	-	0.9806	-
	FedFace	0.4998	-54.7%	0.5865	-59.8%
	FedFR	0.9637	+5.5%	0.9875	+0.7%
	FedFS(Ours)	**0.9794**	**+7.2%**	**0.9934**	**+1.3%**
MobileNetV2	-	0.9339	-	0.9645	-
	FedFace	0.5055	-54.1%	0.5542	-57.4%
	FedFR	0.9588	+2.6%	0.9876	+2.3%
	FedFS(Ours)	**0.9647**	**+3.3%**	**0.9922**	**+2.8%**

Table 1: AUROC of various federated learning methods on DigiFace-1M and VGGFace. Each method uses MobileFaceNet, PocketNet, GhostFaceNet, and MobileNetV2 as a pre-trained model to measure AUROC and the AUROC increase/decrease rate compared to the pre-trained model.

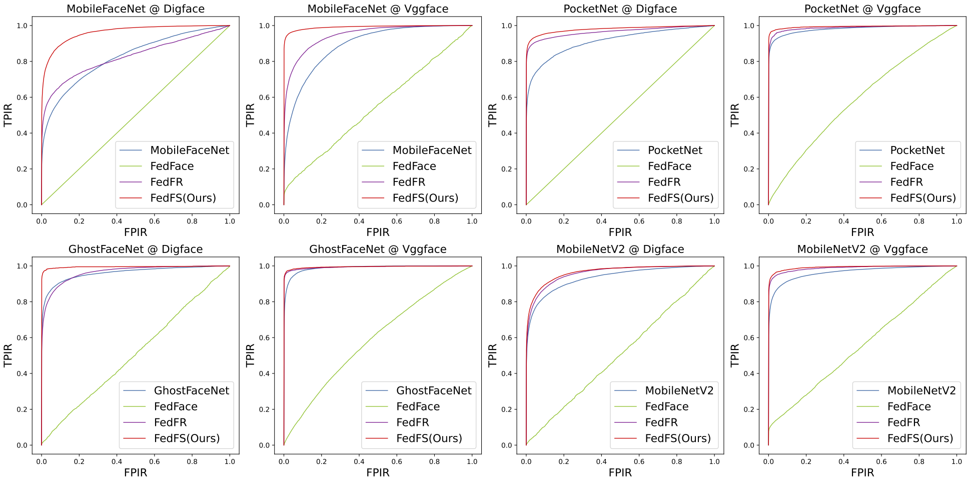


Figure 3: Each graph represents the ROC curve against the pre-trained model @ benchmark and the ROC curve against federated learning methods using the pre-trained model.

4.1 Experiment Setting

We use MS-Celeb-1M [11] to train pre-trained models and share the data publicly in FedFR [20]. And we set MobileFaceNet [7], PocketNet [5], GhostFaceNet [2] and MobileNetV2 [23] as pre-trained models. DigiFace-1M [3] and VGGFace [6] are benchmark datasets for training and evaluation of personalized face recognition models. We use 80% of the images in total for local client training and the remaining 20% of the images for evaluation. Specifically, in each local client, the number of training data and evaluation data are 57/13 for DigiFace-1M and 100/13 for VGGFace, respectively. Also, DigiFace-1M and VGGFace have 10,000 and 8,673 identities, respectively, and each client has only one identity.

In this experiment, we employ 64-layer CNN architecture [21] as a global model and personalized model in the same way as FedFR. We add a linear layer to the last layer for intra-subject self-supervised learning. We use the SGD optimizer with a learning rate of 5e-3. Each client trains 2 local epochs and 5 communication rounds, and clients participating in training are selected randomly.

Participation rates	Method	DigiFace-1M			VGGFace		
		TPIR@FPIR=0.1	TPIR@FPIR=0.01	TPIR@FPIR=0.001	TPIR@FPIR=0.1	TPIR@FPIR=0.01	TPIR@FPIR=0.001
0.01	FedFace	0	0	0	0	0	0
	FedFR	0.4139	0.2133	0.0866	0.7461	0.5464	0.3347
	FedFS(Ours)	**0.6623**	**0.3142**	**0.1397**	**0.9680**	**0.9092**	**0.829**
0.1	FedFace	0	0	0	0	0	0
	FedFR	0.5239	0.2333	0.1113	0.8154	0.5917	0.3513
	FedFS(Ours)	**0.8383**	**0.6057**	**0.3966**	**0.9688**	**0.9149**	**0.8391**
0.3	FedFace	0	0	0	0	0	0
	FedFR	0.6623	0.3066	0.18	0.8308	0.6191	0.3759
	FedFS(Ours)	**0.8603**	**0.6164**	**0.4113**	**0.9765**	**0.925**	**0.8541**
0.5	FedFace	0	0	0	0.0412	0	0
	FedFR	0.7384	0.3666	0.2218	0.9152	0.7474	0.4743
	FedFS(Ours)	**0.8752**	**0.6567**	**0.4586**	**0.9766**	**0.9266**	**0.8586**
0.7	FedFace	0.0912	0.0103	0	0.1231	0.0195	0.0081
	FedFR	0.7451	0.4743	0.3141	0.9315	0.8296	0.6026
	FedFS(Ours)	**0.8905**	**0.6927**	**0.5072**	**0.9786**	**0.9337**	**0.8721**

Table 2: Performance comparison federated learning methods on DigiFace-1M and VGGFace benchmarks. Our proposed method shows the best performance in various participation rates environments.

4.2 Experiment Results

We conduct experiments to analyze how much performance is improved compared to the pre-trained model using FedFace [1], FedFR [20], and our proposed method, FedFS. We use four pre-trained models: MobileFaceNet [7], PocketNet [5], GhostFaceNet [2], and MobileNetV2 [23]. The participation rate of all federated learning algorithms is 0.7, and we calculate AUROC and the percentage improvement based on the pre-trained model. These results are summarized in Table 1, and the ROC Curve graph under the same conditions is shown in Figure 3.

As a result of Table 1 and Fugure 3, we can see that most models show good performance compared to the pre-trained model, but FedFS has the best performance. Because pre-trained models are trained based on large amounts of public data, they are difficult to retrain, and collecting the user data causes privacy issues. Additionally, in the case of the FedFace and FedFR, the performance improvement is not high because they assume a small number of clients and a 1.0 participation rate rather than a large number of clients. On the other hand, our proposed FedFS effectively trains facial features and reduces intra-class variation through intra-subject self-supervised learning using only local data without violating personal information, and shows that significantly improves personalized face recognition performance.

4.3 Performance with various participation rates

Additionally, we compare the performance of the federated learning method using true positive identification rates (TPIR) at different false positive identification rates (FPIR) for 1:N identification protocol [20]. Specifically, we calculate the average TPIR of all clients based on FPIR 0.1, 0.01, and 0.001. The pre-trained model is MobileFaceNet [7], and we set various participation rates: 0.01, 0.1, 0.3, 0.5, and 0.7. According to the experimental results Table 2, the performance of the proposed FedFS shows the best performance in all fields. Through this, FedFS, training using the intra-subject self-supervised learning method, is less affected by the participation rates compared to the previously federated learning methods FedFace [1] and FedFR [20].

5 Conclusion

We proposed FedFS, a federated learning framework to train optimized facial features for each client by using intra-subject self-supervised learning while protecting personal information. Through intra-subject self-supervised learning, we could effectively learn a user's facial features and reduce intra-class variation by simultaneously leveraging dot product and cosine similarities among personal data, resulting in improved recognition performance compared to previous federated learning methods. We believe that FedFS could be applied to various federated face recognition tasks.

References

[1] Divyansh Aggarwal, Jiayu Zhou, and Anil K Jain. Fedface: Collaborative learning of face recognition model. In *2021 IEEE International Joint Conference on Biometrics (IJCB)*, pages 1–8. IEEE, 2021.

[2] Mohamad Alansari, Oussama Abdul Hay, Sajid Javed, Abdulhaid Shoufan, Yahya Zweiri, and Naoufel Werghi. Ghostfacenets: Lightweight face recognition model from cheap operations. *IEEE Access*, 2023.

[3] Gwangbin Bae, Martin de La Gorce, Tadas Baltrušaitis, Charlie Hewitt, Dong Chen, Julien Valentin, Roberto Cipolla, and Jingjing Shen. Digiface-1m: 1 million digital face images for face recognition. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 3526–3535, 2023.

[4] Fadi Boutros, Marco Huber, Patrick Siebke, Tim Rieber, and Naser Damer. Sface: Privacy-friendly and accurate face recognition using synthetic data. In *2022 IEEE International Joint Conference on Biometrics (IJCB)*, pages 1–11. IEEE, 2022.

[5] Fadi Boutros, Patrick Siebke, Marcel Klemt, Naser Damer, Florian Kirchbuchner, and Arjan Kuijper. Pocketnet: Extreme lightweight face recognition network using neural architecture search and multistep knowledge distillation. *IEEE Access*, 10:46823–46833, 2022.

[6] Qiong Cao, Li Shen, Weidi Xie, Omkar M Parkhi, and Andrew Zisserman. Vggface2: A dataset for recognising faces across pose and age. In *2018 13th IEEE international conference on automatic face & gesture recognition (FG 2018)*, pages 67–74. IEEE, 2018.

[7] Sheng Chen, Yang Liu, Xiang Gao, and Zhen Han. Mobilefacenets: Efficient cnns for accurate real-time face verification on mobile devices. In *Biometric Recognition: 13th Chinese Conference, CCBR 2018, Urumqi, China, August 11-12, 2018, Proceedings 13*, pages 428–438. Springer, 2018.

[8] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR, 2020.

[9] Ting Chen, Simon Kornblith, Kevin Swersky, Mohammad Norouzi, and Geoffrey E Hinton. Big self-supervised models are strong semi-supervised learners. *Advances in neural information processing systems*, 33:22243–22255, 2020.

[10] Jiankang Deng, Jia Guo, Niannan Xue, and Stefanos Zafeiriou. Arcface: Additive angular margin loss for deep face recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4690–4699, 2019.

[11] Yandong Guo, Lei Zhang, Yuxiao Hu, Xiaodong He, and Jianfeng Gao. Ms-celeb-1m: A dataset and benchmark for large-scale face recognition. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part III 14*, pages 87–102. Springer, 2016.

[12] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9729–9738, 2020.

[13] Hansol Kim, Youngjun Kwak, Minyoung Jung, Jinho Shin, Youngsung Kim, and Changick Kim. Protofl: Unsupervised federated learning via prototypical distillation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6470–6479, 2023.

[14] Insoo Kim, Seungju Han, Seong-Jin Park, Ji-Won Baek, Jinwoo Shin, Jae-Joon Han, and Changkyu Choi. Discface: Minimum discrepancy learning for deep face recognition. In *Proceedings of the Asian conference on computer vision*, 2020.

[15] Minchul Kim, Anil K Jain, and Xiaoming Liu. Adaface: Quality adaptive margin for face recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 18750–18759, 2022.

[16] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

[17] Youngjun Kwak, Minyoung Jung, Hunjae Yoo, JinHo Shin, and Changick Kim. Liveness score-based regression neural networks for face anti-spoofing. In *ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5, 2023. doi: 10.1109/ICASSP49357.2023.10095535.

[18] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[19] Bi Li, Teng Xi, Gang Zhang, Haocheng Feng, Junyu Han, Jingtuo Liu, Errui Ding, and Wenyu Liu. Dynamic class queue for large scale face recognition in the wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3763–3772, 2021.

[20] Chih-Ting Liu, Chien-Yi Wang, Shao-Yi Chien, and Shang-Hong Lai. Fedfr: Joint optimization federated framework for generic and personalized face recognition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 1656–1664, 2022.

[21] Weiyang Liu, Yandong Wen, Zhiding Yu, Ming Li, Bhiksha Raj, and Le Song. Sphereface: Deep hypersphere embedding for face recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 212–220, 2017.

[22] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Artificial intelligence and statistics*, pages 1273–1282. PMLR, 2017.

[23] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4510–4520, 2018.

[24] Harald Steck, Chaitanya Ekanadham, and Nathan Kallus. Is cosine-similarity of embeddings really about similarity? *arXiv preprint arXiv:2403.05440*, 2024.

[25] Tan Thongtan and Tanasanee Phienthrakul. Sentiment classification using document embeddings trained with cosine similarity. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop*, pages 407–414, 2019.

[26] Nguyen Truong, Kai Sun, Siyao Wang, Florian Guitton, and YiKe Guo. Privacy preservation in federated learning: An insightful survey from the gdpr perspective. *Computers & Security*, 110:102402, 2021.

[27] Hao Wang, Yitong Wang, Zheng Zhou, Xing Ji, Dihong Gong, Jingchao Zhou, Zhifeng Li, and Wei Liu. Cosface: Large margin cosine loss for deep face recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5265–5274, 2018.

[28] Mei Wang and Weihong Deng. Deep face recognition: A survey. *Neurocomputing*, 429:215–244, 2021.