Spatial-Temporal Federated Learning for Lifelong Person Re-identification on Distributed Edges

Lei Zhang, Guanyu Gao, Huaizheng Zhang

Abstract—Data drift is a thorny challenge when deploying person re-identification (ReID) models into real-world devices, where the data distribution is significantly different from that of the training environment and keeps changing. To tackle this issue, we propose a federated spatial-temporal incremental learning approach, named FedSTIL, which leverages both lifelong learning and federated learning to continuously optimize models deployed on many distributed edge clients. Unlike previous efforts, FedSTIL aims to mine spatial-temporal correlations among the knowledge learnt from different edge clients. Specifically, the edge clients first periodically extract general representations of drifted data to optimize their local models. Then, the learnt knowledge from edge clients will be aggregated by centralized parameter server, where the knowledge will be selectively and attentively distilled from spatial- and temporal-dimension with carefully designed mechanisms. Finally, the distilled informative spatial-temporal knowledge will be sent back to correlated edge clients to further improve the recognition accuracy of each edge client with a lifelong learning method. Extensive experiments on a mixture of five real-world datasets demonstrate that our method outperforms others by nearly 4% in Rank-1 accuracy, while reducing communication cost by 62%. All implementation codes are publicly available on https://github.com/MSNLAB/Federated-Lifelong-Person-ReID.

Index Terms—Federated learning, lifelong learning, person re-identification, spatial-temporal knowledge mining.

I. INTRODUCTION

PERSON re-identification (ReID) aims to retrieve people appearing at different locations and moments from the over-lapped cameras. The deep learning-based approaches for person ReID can achieve promising performance on popular benchmarks [1], which enables the applications of person ReID in many computer vision-based applications, such as city surveillance, suspect tracking, and urban analysis.

The deployment of person ReID in real-life still suffers from many great challenges. One prevalent challenge is that the recognition accuracy of the person ReID models will decrease, with the changing of camera environments. This is mainly because of the domain mismatch between the training and deployment environments. Specifically, the person ReID models are usually pre-trained on given datasets which consist of images of a fixed set of person identities captured in specific camera environments. However, in the real-life person ReID deployment, thousands of newly images that are captured every moment often involve many new person identities, which

L. Zhang and G.Y. Gao are with School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing 210094, China. Email: {lei.zhang, gygao}@njust.edu.cn. H.Z. Zhang is with School of Computer Science and Engineering, Nanyang Technological University, Singapore. Email: huaizhen001@e.ntu.edu.sg.
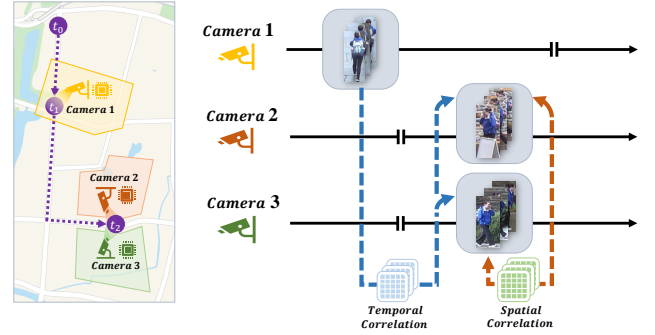

Fig. 1. The spatial-temporal correlations among the data of different edge clients. The image data captured by different cameras have spatial-temporal correlations, and the edge clients can utilize the spatial-temporal knowledge from others for federated learning to continuously improve their performances.

are unavailable at the model training stage. Meanwhile, the camera environments are dynamic and ever-changing due to influences brought by many reasons, such as illumination changing and varying camera views. The domain gap between the training and inference environment limits the performance of person ReID in real-world deployment.

Another challenge for person ReID is to preserve the privacy of the person images [1]. The person images contain sensitive private information, such as individuals' identities, locations, genders, ethnicity, and even facial features [2]. Sharing these sensitive person images for model training and data analytics is infeasible due to the potential risk of privacy leakage and the expensive communication costs for data transmission. Besides, many EU/UK countries have issued privacy protection regulations (e.g., GDPR [3]) to prohibit the centralization of sensitive data from in-situ devices.

To address the domain shift problem for person ReID, some recent works (e.g., [4]–[7]) adopted lifelong learning. These works enable person ReID models to continuously learn knowledge from new scenarios without forgetting previously learnt knowledge. HVIL [5] presents a human-in-the-loop paradigm for lifelong person ReID under interactive manual feedback. CRL [6] and AKA [4] generalise the representation of lifelong person ReID for intra- and inter- domains respectively. GwFReID [7] alleviates forgetting under a class-imbalance lifelong condition for person ReID. However, these works require centralized training with the drifted data from deployed devices to learn new knowledge, which also bring data privacy concerns.

To alleviate privacy concerns, some recent works (e.g., [2], [8], [9]) adopted federated learning to jointly train models

on the edge clients without centralizing data. The federated learning-based approaches enable the sensitive data to be utilized in situ (e.g., [8], [9]), while different edge clients can collaboratively update models by aggregating the gradients or parameters (e.g., [2]) without sharing private data.

Prior works, however, considered the problems of continuously updating models and decentralized training models separately. They are still unable to support distributed edge clients to continuously learn new knowledge while collaboratively sharing their knowledge under privacy-preserving. Hence, we first propose to judiciously combine both federated learning and lifelong learning for person ReID. Moreover, we observe that knowledge learnt from different locations and moments has implicit spatial and temporal correlations. As illustrated in Fig. 1, pedestrians that appeared in the past often reappear on other streets in the near future. We suppose the knowledge learnt from one edge client may also be informative to other neighbor clients shortly. However, previous works neglected the spatial-temporal correlations for recognition knowledge at different locations and moments. Therefore, they failed to adaptively utilize the knowledge across spatial and temporal spaces, and thus limit performance improvement.

We propose a **Fed**erated **S**patial-**T**emporal **I**ncremental **L**earning framework, named FedSTIL, based on spatial-temporal knowledge integration for decentralized continuously learning for person ReID. The edge clients utilize their arriving drift data to optimize the local model for incremental domain knowledge. Meanwhile, some general representations of the drift data are periodically stored in local memory for future rehearsal to alleviate the catastrophic forgetting. Next, the parameter server integrates the incremental knowledge from different edge clients based on the spatial-temporal correlation. Then, the parameter server delivers these informative knowledge to the edge clients. Finally, the edge clients will utilize both the integrated knowledge and previous learnt knowledge to further improve the model with lifelong learning. The framework enables the distributed edge clients to continuously and collaboratively learn from every new scenario without sharing private data.

The main contributions of our paper are summarized as:

- Propose a federated lifelong person ReID framework, which enables the distributed edge clients to continuously learn incremental knowledge with collaboration.
- Design a spatial-temporal knowledge integration method to transfer task-specific knowledge among edge models to improve their performance.
- Demonstrate the effectiveness of our framework via extensive experiments, ablation studies, and visualization.
- Release an open-source tool to facilitate the research for federated lifelong person ReID.

The rests of this paper are organized as follows. Section II introduces the related works, Section III presents the problem definition and the system overview, Section IV illustrates the learning methodology, Section V evaluates the performances of our method, and Section VI concludes the paper.

II. RELATED WORK

In this section, we first introduce some preliminaries of person ReID, and then present the related works about person ReID and federated lifelong learning.

A. Preliminary of Person ReID

Person ReID aims at retrieving a person from non-overlapped camera views. The developments of deep neural network and the large-scale person ReID datasets have significantly improved the performances of person ReID in many vision-based applications [1], [10], [11]. The flows of person ReID mainly includes two stages: training and deployment. The training stage enable person ReID models to learn how to extract the representations from a person's images or videos captured by different non-overlapped cameras. Recent works for training person ReID models have achieved great progress over data processing [12]–[14], modeling [15]–[18], and algorithms [19]–[22]. The goal of the deployment stage is to maintain robust performances for retrieval over the real-life domains. Many recent works for deployment focus on domain adaption [16], [23], [24] and lifelong learning [4]–[7] to narrow the mismatches between different domains.

B. Lifelong Person ReID

In real-world person ReID deployments, camera environments and person characteristics are always different from that of training data. To narrow the domain shifts for the training and deployment stages, some recent works [4]–[7] adopted the lifelong learning strategies for person ReID. Lifelong learning enables the models to continuously learn from every new domain or scenario, which has been widely adopted in many DNN-based serving systems [25]–[27] to deal with the domain drift. The greatest challenge for lifelong person ReID is catastrophic forgetting, which requires ReID models to retain previous knowledge while continually training on new task streams. Recently, HVIL [5] introduced a human-in-the-loop incremental learning method, which enforces models to adaptively refresh and optimize parameters under the supervision of human feedbacks on mismatched person images. AKA [4] proposed a learnable knowledge graph for lifelong person ReID, which can preserve the knowledge from previous domains while propagating learnt knowledge on future unseen domains. GwFReID [7] proposed a class-imbalance lifelong learning for person ReID to generalise the representations for unseen domains without forgetting.

C. Federated Person ReID

With the growing data privacy concern, many person ReID systems are changing to the decentralized or federated training paradigm, where the parameters for edge models can be jointly updated among different source domains instead of centralising private data for training. One challenge for federated person ReID is knowledge interference due to the data heterogeneity across different source domains. To address this problem, FedReID [2] and some other works [8], [9]

adopted model aggregation and knowledge distillation to optimize the performances for the federated person ReID. SKA [9] proposed a selective knowledge aggregation method to transfer personalized knowledge among different edge clients. However, most of the existing works for federated person ReID ignored the spatial-temporal correlation for domain knowledge from different edge locations and moments, which limits the efficiency of knowledge sharing. Our method can capture task-specific knowledge by integrating spatial-temporal knowledge from the edge clients to improve performances.

D. Federated Lifelong Learning

Despite the rapid progresses of lifelong person ReID and federated person ReID, few works studied lifelong person ReID under the federated learning paradigm, which we term as federated lifelong person ReID. Some works (e.g., [28]–[31]) that studied federated lifelong learning are not specific for person ReID, and there are some limitations when applying these methods to person ReID. For instance, FedCurv [28] extended the regularization-based lifelong learning method [32], [33] to the federated learning scenario, and it adopts model aggregation to share knowledge among edge clients. To alleviate catastrophic forgetting during continuously training, it introduces some extra information matrices to prevent the updating of the parameters that are important for previous tasks. However, the high communication cost for transferring the extra information is an obstacle to apply FedCurv [28] in the federated lifelong person ReID scenario. FedWeIT [30] utilized decomposed network layers with local adaptive knowledge and global transferred knowledge. It prevents catastrophic forgetting by replaying the task-specific parameters on inference stages. FedWeIT [30] can be adopted for the incremental task learning where task IDs are provided. However, it is not applicable in the problem scenarios where task IDs are unknown, such as the federated lifelong person ReID scenario. To the best of our knowledge, we are one of the first works which studied federated lifelong learning for person ReID.

TABLE I
MAIN NOTATIONS.

|	
$\mathcal{D}_c^{(t)}$	the drifted training data for edge client c at round t
$\mathcal{P}_c^{(t)}$	the compressed prototypes for raw data $\mathcal{D}_c^{(t)}$
$\overline{\mathcal{P}}_c^{(t)}$	the task feature averaged on prototypes $\mathcal{P}_c^{(t)}$
$\mathcal{G}_c$	the extraction layers for edge client c
$\mathcal{F}_c$	the adaptive layers for edge client c
θ_c	the parameters of adaptive layers for edge client c
A_c	the adaptive parameters with local knowledge
B_c	the base parameters with global spatial-temporal knowledge
α_c	the attention to selectively capture the task-specific knowledge
$\Pi(\cdot)$	the similarity function to measure task features
$\mathcal{S}_{ij}^{(t,t')}$	the similarity between task feature $\overline{\mathcal{P}}_i^{(t)}$ and $\overline{\mathcal{P}}_j^{(t')}$
$W_{ij}^{(t)}$	the knowledge relevance between i- and j-th edge client
$A_c^{(r)}$	the retrieval accuracy of edge client c at the r-th round
$F_c^{(r)}$	the forgetting of edge client c at the r-th round
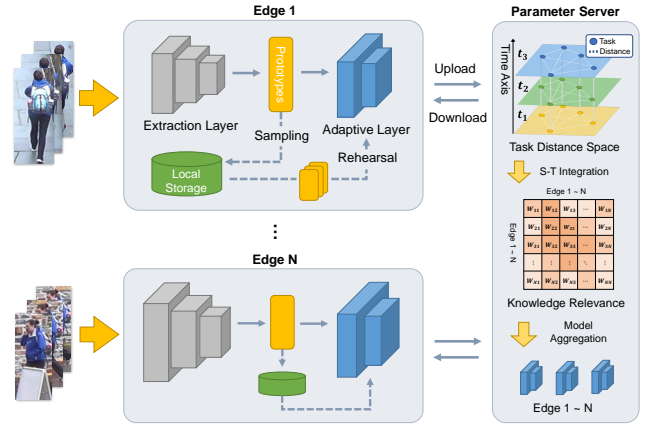
Fig. 2. The architecture of FedSTIL for federated lifelong person ReID. The distributed edge clients continuously learn from both their local drift data and the relevant spatial-temporal knowledge from other edge clients organized by the parameter server to improve recognition accuracy.

III. SYSTEM DESIGN

In this section, we first present the problem definition of the federated lifelong person ReID, and then illustrate the system overview and learning procedure. The main notations of the paper are illustrated in Table I.

A. Problem Definition

We assume that the distributed person ReID system has C edge clients. Each edge client c continuously learns from its arriving task stream $\mathcal{D}_c^{(t)}$, where $\mathcal{D}_c^{(t)}$ denotes the drift data arriving at the t-th round on the edge client c. We assume that $\mathcal{D}_c^{(t)}$ is only available for edge client c on the t-th round, and the previous training data are no longer accessible due to the limited storage space of the edge clients. Our goal is to determine how to continuously learn from both on-edge task streams and across-edge knowledge to improve performances without sharing sensitive raw data among edge clients.

B. System Overview

As illustrated in Fig. 2, our federated lifelong framework, FedSTIL, consists of one central parameter server and several distributed edge clients. The edge clients continuously retrain local models with newly obtained drift data, and the parameter server aggregates spatial-temporal knowledge from the edge clients' local models for knowledge sharing. The network backbones are alternative, such as MobileNet, ResNet, and ViT. We divide the backbone of each edge model into two parts: 1) the extraction layers initialized with global pre-trained weights to extract task prototypes, and 2) the adaptive layers for personalized lifelong learning. Considering the limited computing capacity of the edge clients, our framework adopts the last several layers of the backbone as adaptive layers for training, and the rest layers are extraction layers.

The learning procedures of FedSTIL are as follows. The edge clients collect the new drifted data as incremental tasks and extract the general representation from the drifted data as prototypes of raw data. The edge clients learn incremental
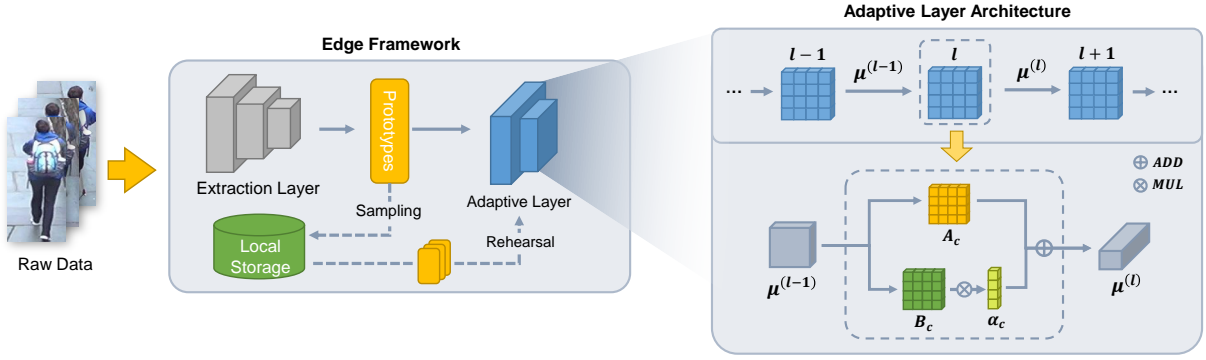
Fig. 3. The architecture of the adaptive layers. The on-edge models continuously learn knowledge from the forthcoming tasks and meanwhile maintain the knowledge from prior tasks. The adaptive layers adaptively balance the tradeoff between local knowledge and global spatial-temporal knowledge.

knowledge from these prototypes of drift data, and then upload learnt knowledge to the parameter server. The parameter server aggregates these incremental knowledge based on the spatial-temporal correlation of neighbors' task characteristics, and delivers the task-relevant knowledge to the edge clients. The edge clients adaptively utilize dispatched knowledge and previously learnt knowledge to optimize models for new scenarios continuously. Meanwhile, some general prototypes of raw data will be stored in each edge client for future rehearsal to alleviate the forgetting of previously learnt knowledge.

IV. LEARNING METHODOLOGY

In this section, we first present the framework of our FedSTIL and then illustrate the training methodology.

A. Lifelong Learning on Distributed Edges

Task Prototype Extraction. The drift data arrives as task stream $\mathcal{D}_c^{(t)}$, where $t = 0, 1, 2, ...$ Considering the limited resources of the edge clients, we utilize the extraction layers to encode the raw tasks $\mathcal{D}_c^{(t)}$ to compressed prototypes to represent the original task for training and inference. Specifically, the drift data of edge client c at the t-th round can be represented as $\mathcal{D}_c^{(t)} = \{(X_i^{(t)}, Y_i^{(t)})\}$, where $X_i^{(t)}$ is the i-th training image of $\mathcal{D}_c^{(t)}$ and $Y_i^{(t)}$ is the corresponding label. We input each raw data $(X_i^{(t)}, Y_i^{(t)}) \in \mathcal{D}_c^{(t)}$ into the extraction layers $\mathcal{G}_c$ to extract prototype as

$$\mathcal{P}_c^{(t)} = \{(\mathcal{G}_c(X_i^{(t)}), Y_i^{(t)})\}, \tag{1}$$

where $\mathcal{G}_c$ is the extraction layers, and $\mathcal{P}_c^{(t)}$ is the extracted prototype set for the raw data set $\mathcal{D}_c^{(t)}$. Hence, the raw task $\mathcal{D}_c^{(t)}$ can be represented as the prototypes $\mathcal{P}_c^{(t)}$ after being processed by the extraction layers. Compared with raw data, prototypes are more generalized to represent different tasks and also smaller to store. Moreover, the prototypes with compressed semantics can also reduce data transmission costs and avoid potential privacy leakage.

Adaptive Lifelong Learning. The training and inference tasks on different edge clients have different characteristics due to the different camera environment. If all of edge clients adopt one single unified model for lifelong training or inference, it is hard to achieve the optimal performance for each edge client.

To address the problem of heterogeneous tasks on different edge clients, each edge client needs to have its personalized model, which can continuously learn from both local knowledge and other edge neighbors' knowledge. To this end, we present the adaptive layers that leverage global and local knowledge for personalized model training. As illustrated in Fig. 3, the parameters θ_c of the adaptive layers for the model of edge client c consist of three parts: 1) the adaptive parameters A_c with knowledge learnt from local incremental tasks, 2) the base parameters B_c with the spatial-temporal knowledge from other correlated edge neighbors, and 3) the attention parameters α_c to capture the task-specific knowledge from the base parameters B_c by attention mechanism,

$$\theta_c = B_c \odot \alpha_c + A_c. \tag{2}$$

As shown in Eq. (2), the parameters θ_c of the adaptive layers are combined by the following terms. The first term $(B_c \odot \alpha_c)$ enables learnable attention α_c for the spatial-temporal knowledge B_c to capture the personalized specific knowledge for local tasks. The second term (A_c) is to learn incremental knowledge from the drifted data of local tasks. We adaptively combine knowledge from two different perspectives into the adaptive layers to continuously optimize models for the newly arriving tasks.
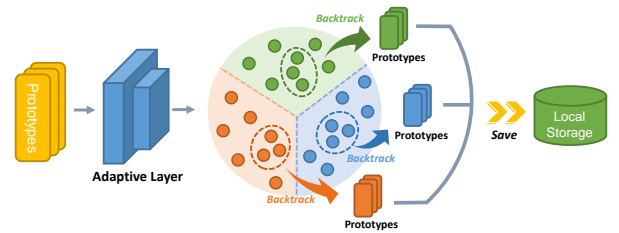

Fig. 4. The data flow for sampling prototypes into local storage. We dynamically store some identities' prototypes which are near the corresponding mean center for future rehearsal.

Prototype Rehearsal to Alleviate Forgetting. The local models of the edge clients are continuously updated with newly arriving tasks. However, the new incremental knowledge will disturb the prior learnt knowledge, and thus the recognition accuracy for the previous domains will decrease during continuous learning [7]. To alleviate catastrophic forgetting of
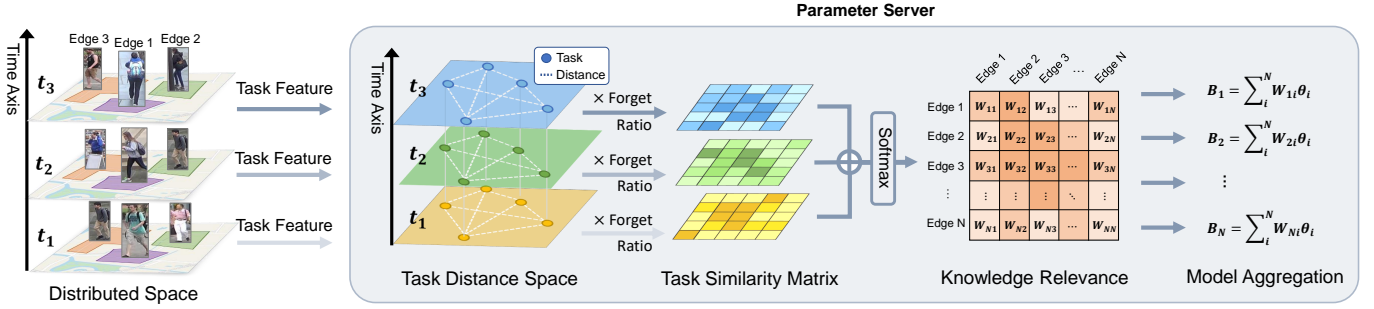
Fig. 5. The framework of the spatial-temporal knowledge integration on the parameter server. The parameter server can automatically measure the spatial-temporal correlations for edge clients based on their task features. The task-specific knowledge is organized for edge clients for continuously learning.

previously learnt knowledge, we design a prototype rehearsal method which utilizes the stored general prototypes of the former tasks in the local storage for knowledge rehearsal.

As shown in Fig. 4, we periodically sample some representative prototypes of the newly arriving incremental task with the nearest-mean-of-exemplars strategy [34], and store them in the local storage of the edge clients. Specifically, when a training task arrives, we first input the prototypes of the task into the adaptive layers. Then, we calculate the mean center of the outputs of the adaptive layers for each person identity. We will store some prototypes whose outputs are closest to the mean centers of different person identities. Those prototypes are informative to maintain the representation of previously learnt knowledge. During the training phases for new tasks, we periodically sample some stored prototypes for training to mitigate the forgetting of the previously learnt knowledge.

Compared with other rehearsal-based lifelong learning methods (e.g., iCaRL [34], and GwFReID [7]), our strategy is more friendly to edge devices with limited storage resources. That is because prototypes are smaller than raw images for storage. Meanwhile, rehearsing the prototypes to alleviate catastrophic forgetting takes smaller computation overhead.

B. Spatial-Temporal Knowledge Integration on Server

The distributed person ReID system consists of many edge clients. However, not all knowledge learnt from edge clients are informative and relevant to others. It may even hinder the training of an edge client if the irrelevant knowledge from others are transferred to the edge client [30]. To extract the task-relevant knowledge from relevant neighbors, we design a spatial-temporal knowledge integration method based on the characteristics of the tasks from different locations and moments. The framework of the spatial-temporal knowledge integration on the parameter server is illustrated in Fig. 5. Next, we elaborate the procedure of spatial-temporal knowledge integration on the parameter server.

Task Similarity across Spatial-Temporal Dimension. The raw data of training task $\mathcal{D}_c^{(t)}$ is encoded into task prototypes $\mathcal{P}_c^{(t)}$, which consist of the compressed semantics of the raw data. To identify the spatial-temporal correlations of the edge tasks, however, directly centralizing and analyzing the raw data $\mathcal{D}_c^{(t)}$ or task prototypes $\mathcal{P}_c^{(t)}$ is consuming for communication,

and exists the potential risk of privacy leakage. Instead, the parameter server only collects the average value of the prototypes as the task feature $\overline{\mathcal{P}}_c^{(t)}$ for different edge tasks,

$$\overline{\mathcal{P}}_c^{(t)} = \frac{1}{|\mathcal{P}_c^{(t)}|} \sum_{p \in \mathcal{P}_c^{(t)}} p, \qquad (3)$$

where $\overline{\mathcal{P}}_c^{(t)}$ is the task feature for the training task on edge client c at the t-th round, $|\mathcal{P}_c^{(t)}|$ is the number of prototypes in the task, and p is the vector of each prototype.

To evaluate the relevance of the tasks across different edge clients and moments, we calculate the task similarity $\mathcal{S}_{ij}^{(t,t')}$ between the tasks of edge client i at the t-th round and edge client j at the t'-th round as

$$\mathcal{S}_{ij}^{(t,t')} = \Pi(\overline{\mathcal{P}}_i^{(t)}, \overline{\mathcal{P}}_j^{(t')}), \qquad (4)$$

where $\Pi(\cdot)$ is the similarity function to measure between task feature $\overline{\mathcal{P}}_i^{(t)}$ and task feature $\overline{\mathcal{P}}_j^{(t')}$. In this work, we adopt Kullback-Leibler Divergence as the similarity function because it can effectively measure the information difference between different distributions.

Knowledge Relevance for Distributed Edge Clients. To organize the task-relevant knowledge for sharing, we calculate the knowledge relevance for distributed edge clients based on their task similarity across the spatial-temporal dimension.

We measure the knowledge relevance by the similarities of all historical tasks for different edge clients. However, the tasks which arrive recently have a more significant impact on the model knowledge, while those coming earlier have less influence because the knowledge learnt from earlier tasks may be forgotten or updated over time. Hence, we introduce the forgetting ratio λ_f ($0 < \lambda_f < 1$) for historical task when calculating the knowledge relevance. Formally, to calculate the knowledge relevance $W_{ij}^{(t)}$ between edge client i and edge client j at the t-th round, we first evaluate the task similarity between the current task on edge client i and the past k tasks on edge client j. Then, we accumulate the task similarities $\mathcal{S}_{ij}^{(t,t')}$ with the forgetting ratio as,

$$W_{ij}^{(t)} = \sum_{t'=t-k}^{t} \lambda_f^{t-t'} \cdot \mathcal{S}_{ij}^{(t,t')}. \qquad (5)$$

Our intuition is that if the historical tasks of edge client j have a higher similarity with the new task on edge client i, then the model knowledge of edge client j has a greater relevance with the new task on edge client i.

Personalized Model Aggregation. The parameter server needs to integrate the task-relevant knowledge for each distributed edge client to improve the recognition accuracy. We adopt the parameter-sharing approach for spatial-temporal knowledge transfer, and the parameters of models are the carrier of spatial-temporal knowledge for transferring. The parameter server will aggregate the parameters of relevant edge models with corresponding knowledge relevance weight $W_{ij}^{(t)}$. Then the aggregated parameters will be dispatched to edge client i as the base parameters B_i for the training,

$$B_i = \sum_{j \in C/i} W_{ij}^{(t)} \cdot \theta_j, \qquad (6)$$

where θ_j are the model parameters of edge client j, and B_i are the task-specific base parameters for edge client i with integrated knowledge from relevant edge clients. The base parameters B_i will be dispatched to edge client i to help the optimization of the local model.

C. Training Methodology

The distributed training procedures of FedSTIL are illustrated in Algorithm 1. The edge clients collect new drifted data as new tasks, and input these tasks into their extraction layers to generate feature vectors as the prototypes. Then, the edge clients upload the task features, which are the average of the prototypes, to the parameter server. The parameter server calculates the similarities with the task features of the historical tasks from different edge clients. Based on these similarities, the parameter server will integrate the task-relevant knowledge as the base parameters for each edge client, and the edge clients can continuously learn from the new tasks based on the integrated knowledge from other edge clients. Finally, when the training of the edge clients' models converges, the parameters of the adaptive layers of the edge models will be uploaded to the parameter server to further improve other spatial-temporal correlated edge clients.

For each task, the edge clients store some of the task prototypes for future rehearsal, which can alleviate forgetting of previously learnt knowledge. Specifically, we periodically select a batch of prototypes from the current task and previously stored prototypes to train the parameters θ_c of the adaptive layers for each task. The loss function can be either cross-entropy loss or triplet loss.

The training samples on the edge clients may be insufficient due to the limited data. If we directly use the limited data to train models, the models may easily fall into overfitting. We adopt parameter tying to tackle this issue by regularizing or penalizing model weights [37], where all parameter changes are summarized as a penalty loss to get sparse gradients for parameters optimization. By tying the parameters of edge clients' models, the models can converge with less overfitting due to the minimal changes of prior knowledge.

Algorithm 1: Training Procedures of FedSTIL

Require: Pre-trained weights $\theta_c^{(0)}$
Require: Task streams $\{\mathcal{D}_c^{(t)}\}_{t=1}^{\infty}$
Require: Shared layers $\{\mathcal{G}_c\}_{c=1}^{C}$
Require: Adaptive layers $\{\mathcal{F}_c\}_{c=1}^{C}$

1 Initialize weights $\{\alpha_c^{(0)}; A_c^{(0)}\}_{c=1}^{C}$
2 **for** *round* $t = 1, 2, 3, ...$ **do**
3   **for** *client* $c \in C$ **do**
4   Collect incremental task $\mathcal{D}_c^{(t)}$ on edge client c
        /* Task Prototype Extraction   */
5   Calculate task prototypes $\mathcal{P}_c^{(t)}$ and $\overline{\mathcal{P}}_c^{(t)}$
        /* Spatial-Temporal Integration   */
6   Calculate task similarity $\{\mathcal{S}_{ci}^{(t,t')}\}_{i \in C/c}$ by Eq.(4)
7   Integrate knowledge relevance $\{W_{ci}^{(t)}\}_{i \in C/c}$ by Eq.(5)
8   Aggregate spatial-temporal knowledge into B_c by Eq.(6)
        /* Adaptive Lifelong Learning   */
9   Set parameters of local adaptive layers $\theta_c \leftarrow B_c \odot \alpha_c + A_c$
        /* Prototypes Rehearsal   */
10   Sample training data X from current and stored prototypes
11   **for** *epoch* $= 1, 2, ...$ **do**
12   Update $\theta_c \leftarrow \theta_c - \eta \nabla \mathcal{L}(\theta_c; X)$
13   Upload parameters θ_c to the parameter server

V. EXPERIMENT

In this section, we present the experimental settings and evaluate the performances of our method by comparing with the baseline methods. We also conduct ablation and visualization studies to validate the effectiveness of our method.

A. Experimental Setting

1) Dataset: We combine the following five ReID datasets to simulate the real-world scenarios of distributed person ReID, namely, Market-1501 [38], PKU-ReID [39], PersonX [40], Prid2011 [41], and DukeMTMC-reID [42]. We shuffle the images of these datasets into 5 distributed edge clients with non-overlapped camera-IDs. Then, the images for each edge client are grouped into 6 sequential tasks. For each task, we randomly sample 60% of the images as training data and 40% of the images as query data. In addition, the gallery images are collected from different edge clients which have different camera views for query images.

2) Backbone Network: The backbone network for person ReID is modified from ResNet-18 [43]. Compared with regular ResNet-18, the differences are as follows: 1) the last stride of ResNet-18 is set to be 1 to enrich the granularity of representations; 2) a batch normalization layer is added after representation to smooth loss for better convergence; 3) the bias of the classifier is removed to avoid representation bias. To

TABLE II
THE PERFORMANCE COMPARISON OF DIFFERENT METHODS.

| Method	Type	mAP (%)	R1 (%)	R3 (%)	R5 (%)	Storage	S2C	C2S
STL	Baseline	54.39	51.62	60.27	63.70	354MB	NaN	NaN
EWC [32]	Lifelong (Regularization)	52.42	49.56	58.34	61.87	154MB	NaN	NaN
MAS [33]	Lifelong (Regularization)	52.32	49.35	58.53	62.31	154MB	NaN	NaN
iCaRL [34]	Lifelong (Rehearsal)	54.52	51.78	60.33	63.63	696MB	NaN	NaN
FedAvg [35]	Federated	62.47	60.26	67.20	69.92	59MB	2.8GB	2.8GB
FedProx [36]	Federated	62.48	60.26	67.30	69.97	106MB	2.8GB	2.8GB
FedCurv [28]	Federated Lifelong	58.26	55.42	64.37	67.75	631MB	30.7GB	6.1GB
FedWeIT (a) [30]	Federated Lifelong	53.86	50.93	59.92	63.67	611MB	3.2GB	1.1GB
FedWeIT (b) [30]	Federated Lifelong	64.06	61.55	68.89	71.38	993MB	8.1GB	2.7GB
FedSTIL (ours)	**Federated Lifelong**	**68.16**	**66.05**	**72.03**	**74.05**	**825MB**	**2.8GB**	**2.8GB**

(a) mAP Accuracy
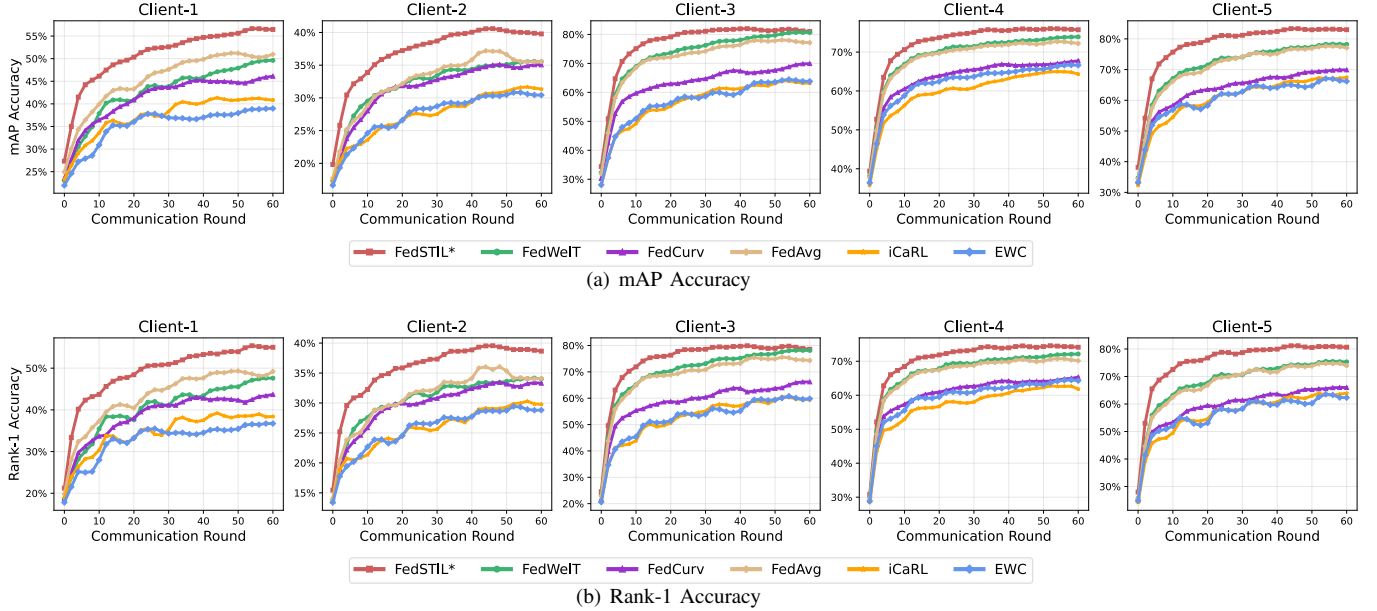
(b) Rank-1 Accuracy

Fig. 6. The comparison of average mAP and Rank-1 accuracy of 6 local tasks on 5 edge clients during 60 communication rounds. Our FedSTIL achieves higher and stable accuracy with less fluctuation during the federated lifelong learning.

verify the compatibility of our method with different backbone networks, we also evaluate the performances with ResNet-50 and Swin-Transformer [44].

3) Training Setting: We adopt Adam optimizer with learning rate 10^{-3} and weight decaying rate 10^{-5}. The edge models are trained for 5 epochs at each communication round between parameter server and edge clients. To avoid the overfitting issue, the training phases will stop if the loss stops decreasing for 3 epochs. We only use random erasing for the image augmentation. Considering the limited computing capacity of the edge devices, only the last residual block and the classifier will be updated during training, and the other layers are fixed with pre-trained weights.

4) Performance Metric: We adopt the following performance metrics to evaluate federated lifelong person ReID, which are in line with the previous works [10], [30].

(1) Accuracy: we adopt the mean average precision (mAP) and cumulative match characteristic (CMC) [10] to measure the retrieval accuracy for each edge client. The test accuracy (i.e., mAP, CMC) at communication round r is defined as the average retrieval accuracy of all training

tasks on edge client c as follows,

$$A_c^{(r)} = \frac{1}{N_c} \sum_{i=1}^{N_c} a(r; \mathcal{D}_c^{(i)}), \quad (7)$$

where $a(r; \mathcal{D}_c^{(i)})$ is the retrieval accuracy of the i-th task $\mathcal{D}_c^{(i)}$ on edge client c at r-th communication round and N_c is the number of training tasks for edge client c.

(2) Forgetting: we measure the forgetting for each client by calculating the decreasing accuracy compared with the maximum value of each task during training [30]. The forgetting of client c can be calculated as follow,

$$F_c^{(r)} = \frac{1}{N_c - 1} \sum_{i=1}^{N_c-1} \max_{t \in \{1,...,r\}} a(t; \mathcal{D}_c^{(i)}) - a(r; \mathcal{D}_c^{(i)}), \quad (8)$$

where $a(r; \mathcal{D}_c^{(i)})$ is the retrieval accuracy as mentioned above. Note that the forgetting for the last training task does not exist because there is no further task for training.
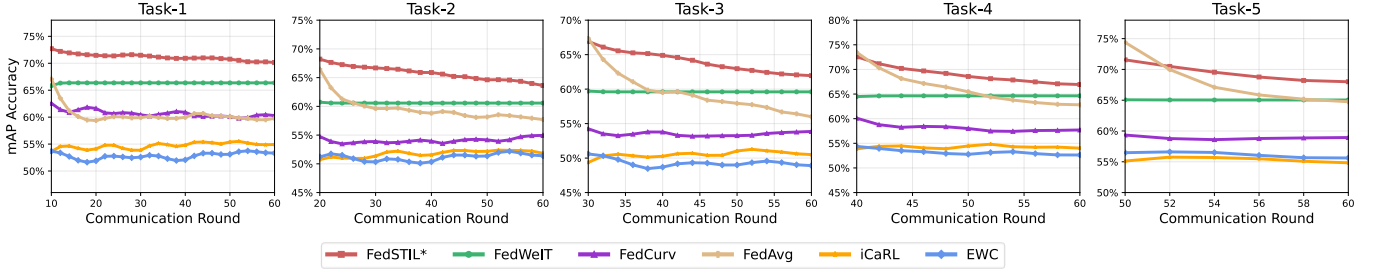
Fig. 7. The comparison of forgetting for trained tasks over different communication rounds. Our FedSTIL can maintain higher accuracy for ever-learnt tasks with less forgetting when training for new tasks.

B. Performance Comparison

We compare the accuracy (mAP, Rank-1, Rank-3, Rank-5), storage cost (model size + memory size), server-to-client (S2C) and client-to-server (C2S) communication cost of our method with the following baseline methods.

1) Single task learning (STL);
2) Lifelong learning: EWC [32], MAS [33], iCaRL [34];
3) Federated learning: FedAvg [35], FedProx [36];
4) Federated lifelong learning: FedCurv [28], FedWeIT [30].

In the real-life deployments of person ReID, task IDs are unavailable both in the training and inference stages. FedWeIT [30] requires task IDs for training and inference, and we assume that task IDs are given for FedWeIT [30] to obtain its performance. Moreover, we validate FedWeIT [30] under different settings to balance its accuracy and communication cost. FedWeIT (a) is set with $l_1 = 1.0 \times 10^{-4}$, $l_2 = 1.0 \times 10^{-6}$. FedWeIT (b) is set with $l_1 = 5.0 \times 10^{-6}$, $l_2 = 1.0 \times 10^{-3}$.

Table II illustrates the performance comparison of different methods. Our FedSTIL can outperform other baseline methods in accuracy and communication cost. The most competitive baseline method, FedWeIT [30], is about 4% lower than our method on mAP, and the S2C communication cost is 2.89 times higher compared with our method. The higher accuracy and lower communication cost make our method more applicable in real-world person ReID scenarios. The detailed analysis for each performance metric are as follows.

1) Comparison of Accuracy: We evaluate the average accuracy of different edge clients under different federated lifelong learning methods (i.e., FedSTIL, FedWeIT [30], FedCurv [28]) in Fig 6. Compared with the baseline methods, our method can achieve higher accuracy over 60 rounds of training. Besides, with continuous training on the new tasks, the accuracy can increase stably with less fluctuation, which also indicates the robustness of our methods. This improvement is largely owing to the effective knowledge sharing and the alleviation of the forgetting of the learnt knowledge during federated lifelong learning. We then analyze the impact of knowledge sharing and forgetting on federated lifelong person ReID.

Knowledge Sharing. Our FedSTIL can achieve higher accuracy because it can effectively exchange knowledge among edge clients. In general, all federated learning-based methods (i.e., FedAvg [35], FedProx [36], FedCurv [28], FedWeIT [30], and our FedSTIL) can achieve higher accuracy than other local-based training methods (i.e., STL, EWC [32], MAS [33],

and iCaRL [34]). These results indicate that exchanging on-edge knowledge can significantly improve the overall accuracy, because the local data of an edge client is limited for learning, such as insufficient data or limited camera views. Thus, sharing the knowledge across edge clients can make up for the limitations of local data. Moreover, we can observe that our FedSTIL achieves the highest accuracy compared with other federated learning-based methods (i.e., FedAvg [35], FedProx [36], FedCurv [28], and FedWeIT [30]). The improvement mainly owes to the task-relevant knowledge sharing among edge clients, which alleviates the interference incurred by the irrelevant knowledge transferring among the edge clients' models.

Alleviation of Forgetting. Our FedSTIL can maintain the accuracy with less forgetting of the previous knowledge during the continuous training on the subsequent tasks. We can observe in Fig. 6 that both the Rank-1 and mAP accuracy of federated learning methods (i.e., FedAvg [35] and FedProx [36]) will not increase after the 40-th round. As shown in Fig. 7, this is because the knowledge learnt from the past tasks is gradually forgotten during the learning of the new tasks. In contrast, federated lifelong methods (i.e., FedCurv [28], FedWeIT [30], and FedSTIL) can alleviate forgetting on the past tasks, which help to achieve higher accuracy.
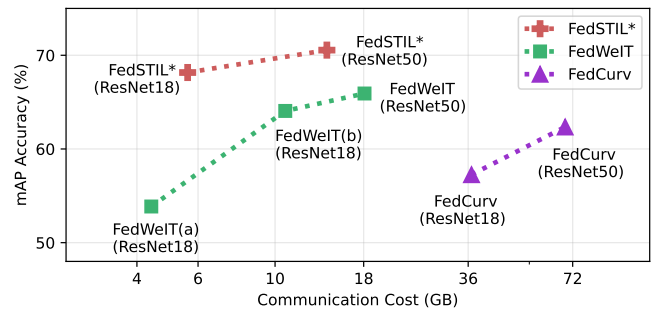

Fig. 8. The comparison of mAP accuracy over communication cost. Our FedSTIL achieves the highest retrieval accuracy with less communication cost.

2) Comparison of Communication Cost: We illustrate the mAP accuracy over total communication costs (client-to-server and server-to-client cost) in Fig. 8. FedSTIL is communication-efficient and can achieve higher accuracy compared with the other federated lifelong learning methods. FedWeIT [30] can adjust the communication cost by l_1 pruning for
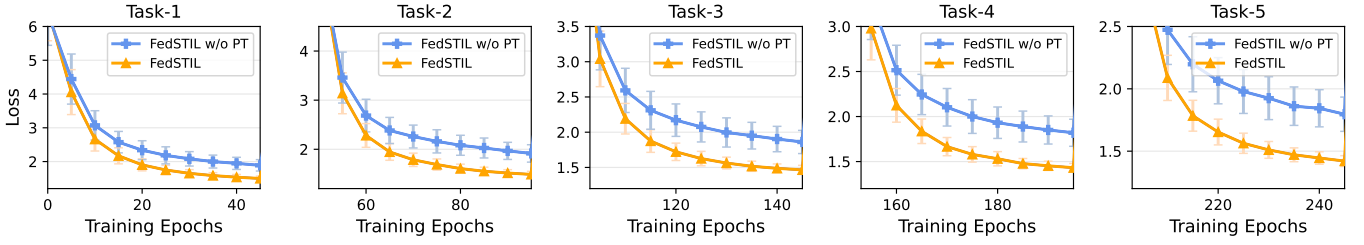
Fig. 9. The comparison of the averaged loss for different tasks w/ and w/o parameter tying. Parameter Tying can achieve convergence faster under continuously training for sequential tasks.

the transferred parameters, however, the accuracy of FedWeIT (a) is still much lower than our method under comparable communication cost. FedCurv [28] has lower accuracy with higher communication cost. That is because FedCurv [28] needs to transfer excessive additional information to alleviate catastrophic forgetting. Our FedSTIL is more communication-efficient because it does not need to exchange additional parameters among edge clients except model weights. This improvement is largely because FedSTIL only requests task-specific parameters, which have been aggregated by the parameter server based on the relevant spatial-temporal knowledge. Moreover, our FedSTIL can achieve higher accuracy compared with the other baseline methods by using the cheaper model ResNet18, which has lower computation cost and storage consumption. Therefore, FedSTIL is more applicable for deployment in real-life person ReID scenarios by reducing communication and computational costs.

C. Ablation Study

We conduct the ablation study by removing some key components of our method to analyze their influences on the performances, and the results are shown in Table III. Specifically, the "w/o S-T Integration" setting is without integrating and transferring spatial-temporal knowledge sharing among edge clients. The "w/o Prototypes Rehearsal" setting does not store prototypes for future rehearsal. The "w/o Parameter Tying" setting removes the tying regularization terms in the loss functions. Table III demonstrates that the accuracy significantly decreases without these components, which verifies that these components contribute to our method's performance improvements. We then conduct detailed ablation studies to verify the effectiveness of each design of our FedSTIL.

TABLE III
THE ABLATION STUDY OF THE INFLUENCES OF SPATIAL-TEMPORAL KNOWLEDGE INTEGRATION, PROTOTYPE REHEARSAL, PARAMETER TYING ON THE ACCURACY OF OUR METHOD.

Variant	mAP (%)	R1 (%)
FedSTIL	**68.16**	**66.05**
w/o S-T Integration	54.26 (-13.90)	51.51 (-14.54)
w/o Prototype Rehearsal	60.73 (-7.43)	58.18 (-7.87)
w/o Parameter Tying	62.53 (-5.63)	60.33 (-5.72)

Influence of Memory Size on Forgetting. We analyze the catastrophic forgetting of our method as Eq. (8) under

TABLE IV
THE COMPARISON OF CATASTROPHIC FORGETTING WITH DIFFERENT MEMORY SIZES FOR PROTOTYPES REHEARSAL.

Variant	Memory	mAP-F ($\downarrow$)	R1-F ($\downarrow$)	R5-F ($\downarrow$)
w/o PR	NaN	5.40	5.71	4.26
+ PR: 10K	415MB	4.87	5.25	3.51
+ PR: 12K	505MB	4.20	4.59	3.19
+ PR: 14K	572MB	4.02	4.36	2.76
+ PR: 16K	662MB	4.14	4.48	2.92
+ PR: 18K	719MB	3.78	4.03	2.83
+ PR: 20K	783MB	3.57	3.81	2.75

different memory sizes for prototype rehearsal. As shown in Table IV, the Rank-1 Forgetting (R1-F), Rank-5 Forgetting (R5-F), and mAP Forgetting (mAP-F) keep decreasing as the memory size increases. With a larger memory size for storing prototypes, the knowledge from more historical tasks can be saved for future rehearsal. Therefore, the rehearsal with more historical prototypes can effectively alleviate the forgetting of the previous knowledge. In addition, the Rank-1 Forgetting with the memory size 20,000 (20K) is nearly 2% lower than without prototype rehearsal, which is within the acceptable range in the real-life person ReID scenario. These results indicate that our prototype rehearsal can effectively alleviate the catastrophic forgetting and retain the knowledge from the previous tasks during federated lifelong learning.

TABLE V
PERFORMANCES WITH DIFFERENT BACKBONES.

Backbone	Method	mAP	R1	R5	Storage	TC
ResNet18	FedCurv	58.26	55.42	67.75	631MB	36.8GB
	FedWeIT	64.06	61.55	71.38	993MB	10.8GB
	FedSTIL	68.16	66.05	74.05	825MB	5.6GB
ResNet50	FedCurv	62.35	58.72	70.03	951MB	68.5GB
	FedWeIT	65.92	62.04	73.58	1.42GB	18.1GB
	FedSTIL	70.55	68.30	75.14	1.62GB	14.1GB
Swin-T	FedCurv	65.71	62.30	72.95	1.14GB	92.2GB
	FedWeIT	66.52	64.35	74.70	1.90GB	22.5GB
	FedSTIL	71.31	69.12	75.03	1.92GB	15.8GB

Performances with Different Backbones. We analyze the influences of choosing different network backbones on the accuracy, storage cost, and communication cost. Specifically, we utilize ResNet18, ResNet50, and Swin-Transformer (Swin-T) [44] as backbones for training. Considering the limited

computing capacity of edge devices, only the last block of these backbones are trainable during learning, and the rest blocks are fixed with the pre-trained weights. As shown in Table V, our FedSTIL can outperform the other baselines on accuracy (i.e., mAP, Rank-1, Rank-5), storage, and total communication cost (TC) with different backbones. The storage size of FedSTIL is comparable with FedWeIT [30] and larger than FedCurv [28]. The communication cost of FedSTIL is lower compared with FedCurv [28] and FedWeIT [30], while the accuracy of FedSTIL is the highest with different backbones. These results verify the compatibility of our method with different backbones. Therefore, one can choose suitable backbones based on the computing capacity of the edge client.

Error Loss without Parameter Tying. To further investigate the effectiveness of parameter tying, we then analyze the error loss of different tasks during training. As illustrated in Fig. 9, we compare the error loss by eliminating the parameter tying from Task-1 to Task-5 during different training epochs. Our FedSTIL with the parameter tying can achieve lower loss and converge much faster than eliminating the parameter tying. The improvement is mainly because parameter tying enables on-edge models to continuously fit new tasks with minimal parameter changes and forgetting of previous knowledge learnt in the past. Thus, the design of tying the spatial-temporal correlated edge models for jointly optimizing can achieve better convergence and generalization.

TABLE VI
PERFORMANCES UNDER DIFFERENT DISTANCE METRICS.

Distance	mAP (%)	R1 (%)	R3 (%)	R5 (%)
Cosine	66.92	65.13	70.92	72.92
Euclidean	67.03	65.27	70.88	72.93
KL	68.16	66.05	72.03	74.05

Performances under Different Distance Metrics. Considering the characteristics of the task features, we adopt Kullback-Leibler (KL) Divergence to compute the spatial-temporal correlation ratio when aggregating the personalized sharing knowledge. We also evaluate the retrieval accuracy of our method under some other distance metrics, namely, Cosine Distance and Euclidean Distance. As shown in Table VI, KL can achieve higher accuracy than other metrics. Different from Cosine and Euclidean, KL can effectively measure the information difference among the different distributions of task features. Hence, KL can help the parameter server precisely measure the edge similarities to aggregate the spatial-temporal knowledge with low noise and less interference.

D. Visualization Study

To further verify the effectiveness of some designs in our method FedSTIL, we visualize the heat-maps of our method by removing spatial-temporal integration and prototype rehearsal to demonstrate their influences on the representation. We utilize the gradient-weighted class activation map (GradCAM) [45] to generate the heat maps, where the hot-spots are the positive focus of the models for re-identification.
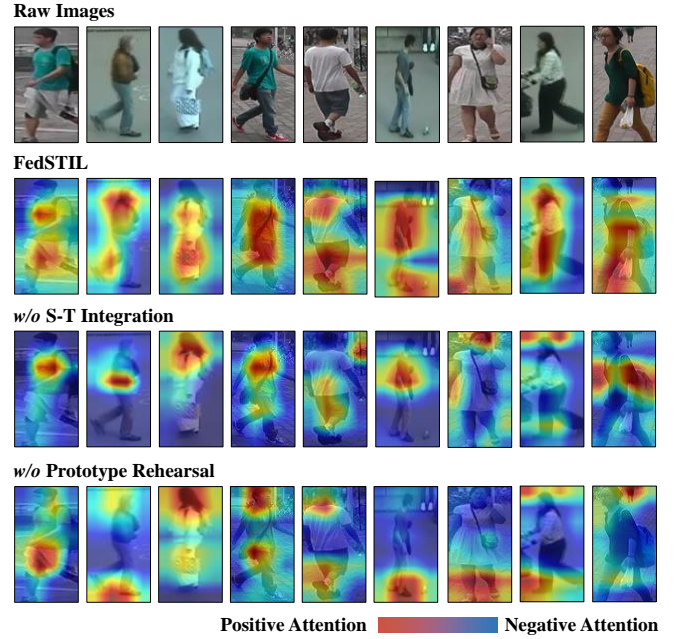

Fig. 10. The heat maps of sampled person images. The hot-spots are what the models focus on, which also reflect the generalization degree of representation. Images with red and blue boundaries denote positive attention and negative attention from the perspective of models.

We can observe in Fig. 10 that without the spatial-temporal knowledge integration, the model will only focus on some limited regions, such as shoes, coats, or bags. On the contrary, the attention regions of our oracle method FedSTIL are mainly on the overall person's body and have more generalized representations. This improvement is largely due to the effectiveness of knowledge sharing to improve the generalizing capacities for edge clients. Moreover, we can also observe that the model without prototype rehearsal may deviate from the regions of a person, and may even focus on the surrounding or background regions. This is probably because the learnt knowledge from the prior tasks will be gradually forgotten during the continuous learning of new scenarios. However, our FedSTIL with the prototype rehearsal will mitigate the forgetting of the former knowledge and maintain the focus on person regions during continuously learning.

VI. CONCLUSION

In this work, we proposed a federated lifelong learning framework, FedSTIL, which enables the distributed edge clients to learn collaboratively and continuously in real-life for person ReID in new scenarios. Our proposed method can utilize the spatial-temporal correlated knowledge among the edge clients to improve the data representation for person ReID and achieve better performance. Meanwhile, the learnt knowledge from the historical tasks can remain long-term effective with only slight forgetting during continuously learning for new scenarios. Experimental results verified that FedSTIL improves the accuracy of person ReID while reducing the communication cost for distributed learning.

REFERENCES

[1] X. Shu, X. Wang, X. Zang, S. Zhang, Y. Chen, G. Li, and Q. Tian, "Large-scale spatio-temporal person re-identification: Algorithms and benchmark," *IEEE Transactions on Circuits and Systems for Video Technology*, 2021.

[2] W. Zhuang, Y. Wen, X. Zhang, X. Gan, D. Yin, D. Zhou, S. Zhang, and S. Yi, "Performance optimization of federated person re-identification via benchmark analysis," in *Proceedings of the 28th ACM International Conference on Multimedia*, 2020, pp. 955–963.

[3] "General data protection regulation," https://gdpr.eu/, European Commission, 2020.

[4] N. Pu, W. Chen, Y. Liu, E. M. Bakker, and M. S. Lew, "Lifelong person re-identification via adaptive knowledge accumulation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 7901–7910.

[5] H. Wang, S. Gong, X. Zhu, and T. Xiang, "Human-in-the-loop person re-identification," in *European conference on computer vision*. Springer, 2016, pp. 405–422.

[6] B. Zhao, S. Tang, D. Chen, H. Bilen, and R. Zhao, "Continual representation learning for biometric identification," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2021, pp. 1198–1208.

[7] G. Wu and S. Gong, "Generalising without forgetting for lifelong person re-identification," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 4, 2021, pp. 2889–2897.

[8] C. Zhang, X. Liu, J. Xu, T. Chen, G. Li, F. Jiang, and X. Li, "An edge based federated learning framework for person re-identification in uav delivery service," in *2021 IEEE International Conference on Web Services (ICWS)*. IEEE, 2021, pp. 500–505.

[9] S. Sun, G. Wu, and S. Gong, "Decentralised person re-identification with selective knowledge aggregation," *arXiv preprint: 2110.11384*, 2021.

[10] Q. Leng, M. Ye, and Q. Tian, "A survey of open-world person re-identification," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 30, no. 4, pp. 1092–1108, 2019.

[11] P. Li, P. Pan, P. Liu, M. Xu, and Y. Yang, "Hierarchical temporal modeling with mutual distance matching for video based person re-identification," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 31, no. 2, pp. 503–511, 2020.

[12] Z. Zheng, L. Zheng, and Y. Yang, "Unlabeled samples generated by gan improve the person re-identification baseline in vitro," in *Proceedings of the IEEE international conference on computer vision*, 2017.

[13] J. Liu, Y. Sun, C. Han, Z. Dou, and W. Li, "Deep representation learning on long-tailed data: A learnable embedding augmentation perspective," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 2970–2979.

[14] R. Takahashi, T. Matsubara, and K. Uehara, "Data augmentation using random image cropping and patching for deep cnns," *IEEE Transactions on Circuits and Systems for Video Technology*, 2019.

[15] L. Qi, L. Wang, J. Huo, Y. Shi, X. Geng, and Y. Gao, "Adversarial camera alignment network for unsupervised cross-camera person re-identification," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 32, no. 5, pp. 2921–2936, 2021.

[16] Z. Zhang, Y. Wang, S. Liu, B. Xiao, and T. S. Durrani, "Cross-domain person re-identification using heterogeneous convolutional network," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 32, no. 3, pp. 1160–1171, 2021.

[17] X. Zhong, T. Lu, W. Huang, M. Ye, X. Jia, and C.-W. Lin, "Grayscale enhancement colorization network for visible-infrared person re-identification," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 32, no. 3, pp. 1418–1430, 2021.

[18] B. Chen, W. Deng, and J. Hu, "Mixed high-order attention network for person re-identification," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, pp. 371–381.

[19] X. Ning, K. Gong, W. Li, L. Zhang, X. Bai, and S. Tian, "Feature refinement and filter network for person re-identification," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 31, no. 9, pp. 3391–3402, 2020.

[20] H. Jin, S. Lai, and X. Qian, "Occlusion-sensitive person re-identification via attribute-based shift attention," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 32, no. 4, pp. 2170–2185, 2021.

[21] A. Hermans, L. Beyer, and B. Leibe, "In defense of the triplet loss for person re-identification," *arXiv preprint arXiv:1703.07737*, 2017.

[22] H. Tan, X. Liu, Y. Bian, H. Wang, and B. Yin, "Incomplete descriptor mining with elastic loss for person re-identification," *IEEE Transactions on Circuits and Systems for Video Technology*, 2021.

[23] H. Li, N. Dong, Z. Yu, D. Tao, and G. Qi, "Triple adversarial learning and multi-view imaginative reasoning for unsupervised domain adaptation person re-identification," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 32, no. 5, pp. 2814–2830, 2021.

[24] S. Li, M. Yuan, J. Chen, and Z. Hu, "Adadc: Adaptive deep clustering for unsupervised domain adaptation in person re-identification," *IEEE Transactions on Circuits and Systems for Video Technology*, 2021.

[25] Y. Huang, H. Zhang, Y. Wen, P. Sun, and N. B. D. Ta, "Modelci-e: Enabling continual learning in deep learning serving systems," *arXiv preprint arXiv:2106.03122*, 2021.

[26] H. Zhang, Y. Li, Y. Huang, Y. Wen, J. Yin, and K. Guan, "Mlmodelci: An automatic cloud platform for efficient mlaas," in *Proceedings of the 28th ACM International Conference on Multimedia*, 2020.

[27] H. Zhang, M. Shen, Y. Huang, Y. Wen, Y. Luo, G. Gao, and K. Guan, "A serverless cloud-fog platform for dnn-based video analytics with incremental learning," *arXiv preprint arXiv:2102.03012*, 2021.

[28] N. Shoham, T. Avidor, A. Keren, N. Israel, D. Benditkis, L. Mor-Yosef, and I. Zeitak, "Overcoming forgetting in federated learning on non-iid data," *arXiv preprint arXiv:1910.07796*, 2019.

[29] X. Yao and L. Sun, "Continual local training for better initialization of federated models," in *2020 IEEE International Conference on Image Processing (ICIP)*. IEEE, 2020, pp. 1736–1740.

[30] J. Yoon, W. Jeong, G. Lee, E. Yang, and S. J. Hwang, "Federated continual learning with weighted inter-client transfer," in *International Conference on Machine Learning*. PMLR, 2021, pp. 12073–12086.

[31] S. M. Hendryx, D. R. KC, B. Walls, and C. T. Morrison, "Federated reconnaissance: Efficient, distributed, class-incremental learning," *arXiv preprint arXiv:2109.00150*, 2021.

[32] J. Kirkpatrick, R. Pascanu, N. Rabinowitz, J. Veness, G. Desjardins, A. A. Rusu, K. Milan, J. Quan, T. Ramalho, A. Grabska-Barwinska *et al.*, "Overcoming catastrophic forgetting in neural networks," *Proceedings of the national academy of sciences*, 2017.

[33] R. Aljundi, F. Babiloni, M. Elhoseiny, M. Rohrbach, and T. Tuytelaars, "Memory aware synapses: Learning what (not) to forget," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018.

[34] S.-A. Rebuffi, A. Kolesnikov, G. Sperl, and C. H. Lampert, "icarl: Incremental classifier and representation learning," in *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2017.

[35] J. Konečnỳ, H. B. McMahan, F. X. Yu, P. Richtárik, A. T. Suresh, and D. Bacon, "Federated learning: Strategies for improving communication efficiency," *arXiv preprint arXiv:1610.05492*, 2016.

[36] T. Li, A. K. Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar, and V. Smith, "Federated optimization in heterogeneous networks," *Proceedings of Machine Learning and Systems*, vol. 2, pp. 429–450, 2020.

[37] D. Zhang, H. Wang, M. Figueiredo, and L. Balzano, "Learning to share: Simultaneous parameter tying and sparsification in deep learning," in *International Conference on Learning Representations*, 2018.

[38] L. Zheng, L. Shen, L. Tian, S. Wang, J. Wang, and Q. Tian, "Scalable person re-identification: A benchmark," in *Computer Vision, IEEE International Conference on*, 2015.

[39] L. Ma, H. Liu, L. Hu, C. Wang, and Q. Sun, "Orientation driven bag of appearances for person re-identification," *arXiv preprint arXiv:1605.02464*, 2016.

[40] X. Sun and L. Zheng, "Dissecting person re-identification from the viewpoint of viewpoint," in *CVPR*, 2019.

[41] M. Hirzer, C. Beleznai, P. M. Roth, and H. Bischof, "Person re-identification by descriptive and discriminative classification," in *Scandinavian conference on Image analysis*. Springer, 2011, pp. 91–102.

[42] E. Ristani, F. Solera, R. Zou, R. Cucchiara, and C. Tomasi, "Performance measures and a data set for multi-target, multi-camera tracking," in *European Conference on Computer Vision workshop on Benchmarking Multi-Target Tracking*, 2016.

[43] H. Luo, Y. Gu, X. Liao, S. Lai, and W. Jiang, "Bag of tricks and a strong baseline for deep person re-identification," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, 2019, pp. 0–0.

[44] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo, "Swin transformer: Hierarchical vision transformer using shifted windows," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 10012–10022.

[45] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-cam: Visual explanations from deep networks via gradient-based localization," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 618–626.