This WACV workshop paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

FedFSLAR: A Federated Learning Framework for Few-shot Action Recognition

Nguyen Anh Tu *†1, Assanali Abu†1, Nartay Aikyn[1], Nursultan Makhanov[1], Min-Ho Lee[1], Khiem Le-Huy[2], and Kok-Seng Wong[2]

[1]Department of Computer Science, School of Engineering and Digital Sciences, Nazarbayev University, 53 Kabanbay Batyr Ave., Astana, Kazakhstan, 010000
[2]College of Engineering and Computer Science, VinUniversity, Hanoi, Viet Nam

Abstract

In recent years, Federated Learning (FL) has emerged as a promising solution for many computer vision applications due to its effectiveness in handling data privacy and communication overhead. However, when applying FL to advanced and computationally heavy tasks like video-based action recognition, FL clients can struggle with the lack of annotated data and model biases, thus negatively impacting learning performance. Therefore, adopting Few-Shot Learning (FSL) is essential, where the learned model can adapt to unseen classes using limited labeled examples. Nonetheless, FSL has rarely been exploited for vision tasks under FL settings. In this paper, we develop a Federated Few-Shot Learning framework, FedFSLAR, that collaboratively learns the classification model from multiple FL clients to recognize unseen actions with a few labeled video samples. Prior works in few-shot action recognition mostly use 2D-CNNs as feature backbones and ineffectively capture the temporal correlation between video frames. To overcome this limitation and enable more robust representation, we integrate the spatiotemporal feature backbones based on 3D-CNNs into a meta-learning paradigm, i.e., ProtoNet. Accordingly, we conduct extensive experiments under practical FL settings, e.g., non-IID data, to evaluate various 3D-CNN models alongside representative FL algorithms, i.e., FedAvg and FedProx. Experimental results on benchmark datasets validate the effectiveness of our FedFSLAR framework. Remarkably, our findings indicate that combining feature backbones pre-trained on external data with the FL setting can incredibly benefit FSL. Our framework offers a viable path toward achieving notable progress in FL and FSL for action recognition tasks.

1. Introduction

Thanks to advanced Deep Learning (DL) architectures and massive datasets, recent years have witnessed remarkable progress in video-based human action recognition (HAR) [17], which has many applications in intelligent surveillance. For example, classifying human actions from CCTV cameras is often crucial for security applications. However, training action models to learn robust features typically requires a lot of labeled video data collected from numerous sources with substantial computing resources. Current DL-based methods [5, 36] mainly rely on centralized training, which demands high storage costs and communication overheads to transmit local data from the clients to the central server, thus limiting their practical applicability. Moreover, human bodies from action videos or CCTV streams usually reveal much person-related information, such as personal identity, gender, age, and motion patterns. Such identifiable information can be easily exploited without users' consent for various analysis purposes, leading to privacy breaches. Therefore, Federated Learning (FL) [19] has emerged as an effective way to enable decentralized training, where a shared model can be learned collaboratively while keeping the data on distributed devices. Specifically, FL aggregates and coordinates the local models computed on each device to train the globally shared model on the central server. In this way, privacy protection and communication efficiency are improved without sending sensitive video data to the server. By effectively addressing privacy concerns, FL allows us to train the models using considerably more varied and diverse datasets from heterogeneous devices, thereby learning more complex patterns.

Like centralized learning, FL requires a vast volume of labeled data, e.g., hundreds of samples per class, to achieve the desirable training performance of deep neural networks [16]. However, manually annotating abundant videos is highly expensive and tedious. Moreover, it is problematic

[*]Corresponding author: tu.nguyen@nu.edu.kz
[†]Equal contribution

for end users or organizations, e.g., airports, schools, banks, and hospitals, to compile new video datasets whenever dealing with new action classes. In many scenarios, novel action classes corresponding to new tasks continuously appear over time, and it is difficult to collect data with sufficient annotations to solve these tasks. Reasonable preparation of the new training dataset is very inconvenient due to the diversity of human actions. Also, FL frameworks are constructed based on the assumption that each client must have enough training data for specific tasks. In practice, video data from heterogeneous devices is usually incomplete, drastically restricting the applicability and scalability of FL. In addition, some organizations might have only a few video samples for specific classes, while others might own many more data samples on their local machines. This can make FL struggle with model bias and decrease generalizability. Therefore, to overcome these limitations, Few-Shot Learning (FSL) [32] has emerged as a promising solution to train machine learning models to recognize novel action classes with few support samples.

The typical approach to solving the FSL problem is to use meta-learning, which aims to imitate the learning ability of humans by leveraging prior knowledge and experiences gained from previous tasks. More precisely, meta-knowledge is acquired by learning numerous action videos from base classes and generalized to novel classes with few labeled videos. It is important to note that base and novel classes must be distinct. Current methods [39, 38] mainly employ 2D-CNN backbones to extract frame-level features and then adopt metric learning with temporal alignment techniques to measure the similarity between different videos for classification. Although these methods have achieved significant progress in few-shot video classification, they are only suitable for centralized learning on a single machine. Also, using frame-level features is ineffective in capturing the temporal correlation among video frames. In contrast, we aim to design an FL framework to meta-train robust spatiotemporal deep models using multiple data sources on distributed devices. This problem can be referred to as Federated Few-Shot Learning [8], which takes advantage of FL to provide privacy protection and reduce communication costs while improving the practicality of FL. However, performing FSL in FL environments is challenging due to the discrepancy in task domains across different types of clients.

In this paper, we propose a **Fed**erated **F**ew-**S**hot **L**earning for **A**ction **R**ecognition framework, namely FedFSLAR, for human action recognition tasks. Specifically, given few-shot tasks generated from local videos, we first train meta-learner on each client using an FSL algorithm, Prototypical Networks [25]. Then, local models from different meta-learners are sent to the federated server to aggregate and update the global model. Moreover, to overcome the limitation of existing works based on 2D-CNN feature backbones, we investigate the effectiveness of spatiotemporal deep networks, e.g., 3D-CNN [5, 10], for feature embedding with the capability of exploring the temporal correlation among consecutive frames. A good feature embedding can enable us to learn strong video representations to address the issue of different task domains across heterogeneous clients. To the best of our knowledge, this is the first study exploring Federated Few-Shot Learning for video-based action recognition.

Using the FedFSLAR framework, we provide the benchmark to support more investigation into the field and enable fair comparison. The datasets used in our experiments are divided into identical and independent distribution (IID) and non-identical and independent distribution (non-IID) to achieve realistic FL settings. Through extensive experiments on benchmark datasets, we show that discriminative video representations, which can be transferred and fine-tuned in new tasks with novel classes, effectively generalize meta-knowledge and reduce the gap between centralized learning and FL. Specifically, when feature backbones are pre-trained on the external data, FL can achieve comparable or even higher accuracy than centralized learning for FSL tasks. However, training feature backbones from scratch with random weights will result in a significant drop in accuracy. These results indicate that several issues warrant further investigation: robust representation learning demands effective designs of pre-training schemes for few-shot action recognition; learning meta-knowledge from video data with less variation is challenging, particularly in the non-IID setting where clients have a few classes. These issues open further research directions on FL and FSL, especially for action recognition tasks. Our main contributions are summarized as follows:

- We propose a unified FL framework for few-shot action recognition.
- We comprehensively carry out an empirical study of various 3D-CNNs as feature backbones. We especially investigate the impact of pre-training on the federated FSL performance.
- We systematically compare two popular FL algorithms, e.g., FedAvg[22] and FedProx[19], under various realistic settings for the federated FSL.
- Our benchmark reveals that combining pre-trained 3D-CNNs with suitable FL settings can achieve state-of-the-art performance on challenging datasets of few-shot action recognition.

2. Related Works

2.1. Few-Shot Action Recognition

Few-shot action recognition is a task that recognizes new actions from a few examples. The primary methods

[38, 4, 24, 31] are metric-based meta-learning, which learns a generalizable metric space to compare videos of different actions. These methods typically employ 2D-CNNs to extract frame-level features. Then, the distance between the query and support videos is measured using these features to assess similarity. For example, [38] proposed a compound memory network, CMN, structure with multiple constituent keys and a multi-saliency embedding algorithm. Cao et al. introduced OTAM [4], an online temporal attentive module that learns to select and align the most informative frames from the support set. TRX [24] used CrossTransformers to make class prototypes from relevant sub-sequences of support videos and compare video tuples of different frame numbers. Wang et al. [31] proposed HyRSM, a hybrid relation score module combining two types of relation scores based on global and local features.

Apart from using image-level extractors, several methods [33, 2, 21] employed spatiotemporal models to generate video-level features. For example, TSL [33] used R(2+1)D [27] as the feature backbone and utilized label text queries to retrieve additional videos to augment the sample set data. TARN [2] and CMOT [21] used C3D [26] to extract features and compare variable-length videos with deep distance. Other works [18, 12] attempted to create more data samples by either data augmentation or generating new samples by generative models. ProtoGAN [18] adopted a conditional GAN to produce more samples for each class in the support set. Diferently, AMeFu-Net [12] combined depth and visual information using temporal asynchronization augmentation. Despite the remarkable success, these methods are solely centralized. None of them applied FL for few-shot action recognition, hence inducing severe privacy concerns and communication costs. As a result, their practicality is limited significantly.

2.2. Federated Action Recognition

Federated action recognition is pioneering privacy-focused approaches in computer vision. It uses FL and distributed training to protect sensitive data. This paradigm shift replaces centralized data accumulation with edge devices or servers to maintain privacy while building a global model. The decentralized method is crucial in privacy-focused, bandwidth-limited contexts with limited bandwidth and is widely used in surveillance, healthcare, and other areas where privacy is a priority. Federated action recognition significantly advances secure and efficient action recognition models. For instance, distracted driver activities were studied by [7, 37]. Zhang et al. [37] and Doshi et al. in [7] researched the application of FedAvg [22] with 2D-CNN models on video to detect distracted driver activities. The latter work applied group knowledge transfer algorithm FedGKT [14] additionally to minimize resource usage on edge devices. Xiao et al. [34] researched FL for wearable sensor-based human action recognition (HAR). The authors developed a hybrid CNN, LSTM-based attention algorithm for FedAvg with homomorphic encryption by outperforming existing HAR algorithms on different datasets. Another work by [23] applied FL for HAR application. Ouyang et al. proposed a clustering-based FL system (ClusterFL) that utilizes the clustering relationship among users' data to enhance model accuracy and communication efficiency.

2.3. Federated Few-Shot Learning

Recently, few-shot learning (FSL) has been studied under the FL settings to resolve the problem of limited annotated training data that participating clients own. Fan et al. [8] first introduced this particular problem and proposed formulating the training adversarially and optimizing the client models to produce a discriminative feature space that can better represent unseen data samples. Next, Wang et al. [29] proposed a novel federated few-shot learning (FedFSL) framework with two separately updated models and dedicated training strategies to reduce the adverse impact of global data variance and local data insufficiency. From the application perspective, Chen et al. [6] introduced the FedMeta-FFD framework to enhance mechanical fault diagnosis in the industrial Internet of Things. Cai et al. [3] presented FeS, a framework that enables practical few-shot NLP fine-tuning on federated mobile devices. Besides that, Hoang et al. [15] designed a novel framework, termed F2LCough, to solve the cough sound classification for diagnosing and treating respiratory diseases. Although these works have studied FedFSL for several simple tasks with different data modalities such as image, text, or audio, its application to more advanced vision tasks like video-based action recognition is still unexplored.

3. Methodology

3.1. Problem Definition and Overall Pipeline

Federated few-shot learning aims to recognize classes with limited examples from a set of clients $\{\mathcal{C}_k\}_{k=1}^{K}$ and a federated server $\mathbb{S}$. In the client $\mathcal{C}_k$, FSL divides classes into two disjoint sets: base classes ($\mathcal{X}_b^{(k)}$) used for training with abundant labeled samples, and novel classes ($\mathcal{X}_n^{(k)}$) with a few samples that the model has never encountered before. Then, the meta-learning approach is formulated as an episodic training, where episodes in the client $\mathcal{C}_k$ are sampled from $\mathcal{X}_b^{(k)}$. Subsequently, each episode relevant to a specific task consists of a support set $S^{(k)} = \{(\mathbf{x}_i, y_i)\}_{i=1}^{M\times P}$ and a query set $Q^{(k)} = \{\mathbf{x}_j\}_{j=1}^{L}$. Here, the support set contains labeled examples from M different classes, with P samples per each class. The query set contains L unlabeled samples used for classification within this episode. The problem can be defined as M-way P-shot.

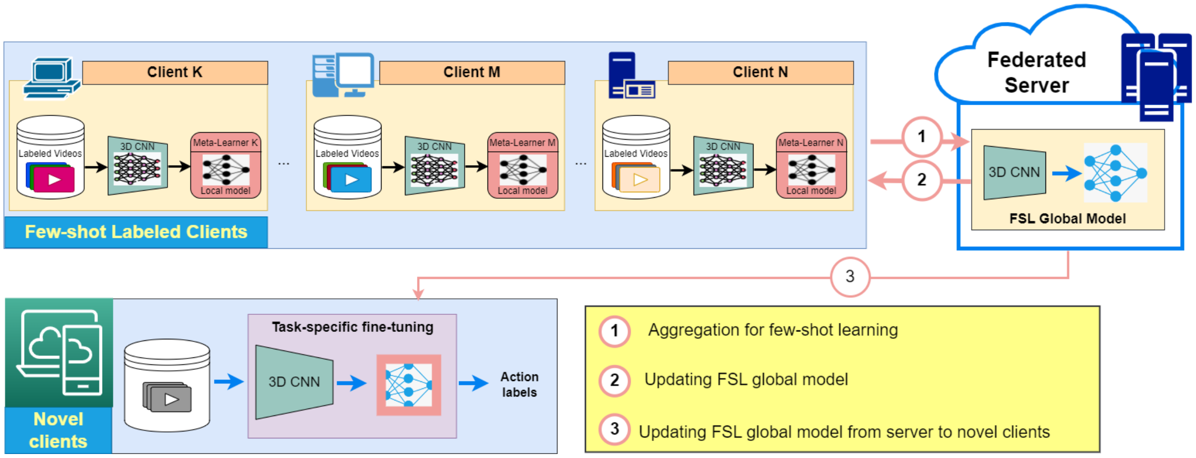


Figure 1. Overall System Pipeline for Federated Few-shot Action Recognition.

Given an episode of the client $\mathcal{C}_k$, the objective is to train a model $\Theta^{(k)}$ to classify the query set $Q^{(k)}$ using the information provided in the support set $S^{(k)}$. This can be done by minimizing the loss between the predicted labels of $Q^{(k)}$ and the true labels. Under the FL setting, the local models $\{\Theta^{(k)}\}$ are transferred to the federated server for aggregating the global model $\{\Theta\}$, which will then be broadcast to clients in the subsequent rounds. During testing, we employ the learned model Θ to fine-tune new tasks in the novel clients. The overall architecture of our proposed method is illustrated in Fig. 1.

3.2. Few-shot Learning

Our FedFSLAR framework uses the Prototypical Networks (ProtoNet)[25] algorithms to perform meta-learning on the base classes of multiple clients. ProtoNet aims to compute a prototype representation for each base class (i.e., the mean of the feature vectors in the same class). As a metric-based classifier, the model is trained to estimate the similarity between examples from the support set and the query during the meta-learning process. To simplify the notation, we consider a single client in this subsection as the meta-learning algorithm is identical for all federated clients. Specifically, the ProtoNet takes an episode that contains a support set $\mathcal{S}$ and an unlabeled query sample $\mathbf{q}$ and performs the following steps.

First, the algorithm obtains the prototype representation $\mathbf{p}_c$ for each class c in the episode by averaging the feature representations of the support samples from that class:

$$\begin{aligned} \mathbf{p}_c &= \frac{1}{N_c} \sum_{(\mathbf{x},y)\in\mathcal{S}_c} f(\mathbf{x}) \\ \mathcal{S}_c &= \{(\mathbf{x}, y) \in \mathcal{S} : y = c\} \end{aligned} \tag{1}$$

where $\mathcal{S}_c$ is the set of all the support samples for class c, $\mathbf{x}$ is a sample, y is its corresponding class label, and $f(\cdot)$ is the feature extractor model.

Next, a distance metric (e.g., Euclidean distance) is used to calculate the similarity between $\mathbf{q}$ and $\mathbf{p}_c$.

$$d(\mathbf{q}, \mathbf{p}_c) = \|f(\mathbf{q}) - \mathbf{p}_c\| \tag{2}$$

Finally, the algorithm applies the softmax function over the negative distances between the query sample and the prototypes to get the predicted class probability for a query sample $\mathbf{q}$ belonging to class c:

$$P(y = c|f(\mathbf{q}), \mathcal{S}) = \frac{\exp(-d(\mathbf{q}, \mathbf{p}_c))}{\sum_{c'} \exp(-d(\mathbf{q}, \mathbf{p}_{c'}))} \tag{3}$$

The ProtoNet predicts the class probabilities for each query sample. These are used to train f with the cross-entropy loss on base classes during the meta-training stage or to evaluate the model on novel classes during the meta-testing stage.

3.3. Federated Learning

In this work, we employ two popular FL algorithms to facilitate thorough comparisons for few-shot action recognition. These algorithms are described as follows.

FedAvg [22] is a foundational FL algorithm designed to train a global model across decentralized edge devices while preserving data privacy. The FL process begins by partitioning a large dataset among client devices, where each client performs local model updates on its respective data partition to minimize the local loss function. The updated models are then sent to a central server, which uses a weighted average to aggregate these local models. The update rule for every client can be expressed as:

$$\Theta_{t+1} = \Theta_t - \eta \nabla g_t(\Theta_t) \tag{4}$$

where Θ_t represents the model parameters at iteration t, $g_t(\Theta_t)$ is the local loss function, ∇ denotes the gradient operator, and η represents the learning rate.

FedProx [19] is an extension of FedAvg that introduces a proximal term to the optimization goal. The aim is to mitigate the impact of non-IID data distributions, which can lead to slower convergence in FL. The proximal term is a regularization component that causes the model parameters to remain close to their previous values. The FedProx update rule can be represented as follows:

$$\Theta_{t+1} = \arg\min_{\Theta} \left[\frac{1}{K} \sum_{k=1}^{K} g_i(\Theta) + \frac{\mu}{2} \|\Theta - \Theta_t\|^2 \right] \quad (5)$$

where K is the number of clients, $g_i(\Theta)$ is the local loss function on the client k, μ is the proximal term weight, and $\| \cdot \|$ represents the Euclidean norm. The proximal term ensures that the updated model remains close to the previous solution, promoting stability and convergence in scenarios with non-IID data distributions.

3.4. Federated Few-shot Learning Framework

The procedure of our proposed FedFSLAR is summarized in Algorithm 1. FedSLAR is an FL approach tailored for learning tasks in a collaborative multi-client setting. Operating over communication rounds, this algorithm orchestrates the synchronized optimization of individual client models. Each client begins by sampling episodes from its local dataset. The clients then enhance their local models using episodic training processes akin to ProtoNet. Following these local model updates, client models are aggregated to form a new global model by considering the proportion of local data held by each client.

Algorithm 1: FedSLAR Algorithm

```
Input      : Communication rounds $T$, Number
             of clients $K$, Dataset $(\mathcal{X}_b, \mathcal{X}_n)$,
             Batch Size $B$, Local Epochs $E$, and
             Learning Rate $\eta$
Output     : Global Model $\Theta$
Initialize : $\Theta_0$
for $t \leftarrow T$ do
    for each client $k$ in parallel do
        $\Theta_t^{(k)} \leftarrow \mathbf{ClientUpdate}(\Theta_t)$
    end
    Clients send model parameter $\{\Theta_t^{(k)}\}_{k=1}^{K}$ to
    server for aggregation:
    $\Theta_{t+1} = \sum_{k=1}^{K} \frac{|\mathcal{X}_b^{(k)}|}{|\mathcal{X}_b|} \Theta_t^{(k)}$
end
Return $\Theta_t$
ClientUpdate():
  Input: global model from previous round $\Theta_t$
  Output: updated local model $\Theta_t^{(k)}$
  Sample a set of episodes from $\mathcal{X}_b^{(k)}$:
    $\mathcal{B}_k = \{\mathcal{T}_1, .., \mathcal{T}_m\}$
  Optimize $\Theta_t^{(k)}$ using episodic training process
like ProtoNet
  Return $\Theta_t^{(k)}$
```

4. Experiments

4.1. Datasets

To evaluate the performance of our framework, we perform experiments on two benchmark datasets: Kinetics [5] and Something-Something-V2 (SSv2)[13]. Something-Something-V2 (SSv2) contains 100 classes divided into 64 non-overlapping classes for the meta-training set, 12 for the meta-validation set, and 24 for the meta-testing set. Moreover, we use a subset of the Kinetics dataset with 100 classes, namely Kinetics-100 (K100), presented in [38] to assess the performance of few-shot action recognition models. Like SSv2, we select 64, 12, and 24 non-overlapping classes for the meta-training, meta-validation, and meta-testing, respectively.

4.2. 3D-CNN models

We employ three 3D-CNN models as feature backbones for FedFSLAR, including R3D-18 [27], I3D [5], and SlowFast [11]. For SlowFast, we particularly used a Slow pathway with 8 frames as input. In our experiments, we use the available codes of these models and their pre-trained weights from Pytorchvideo [9] and Torchvision[1]. Note that different from a majority of few-shot action recognition methods whose feature backbones are 2D-CNNs or Vision Transformer[20] and are pre-trained on ImageNet, our spatiotemporal feature backbones are pre-trained on K400[5].

4.3. Implementation Details

4.3.1 FSL settings

We used the data augmentation and video preprocessing strategy described in TSN [28] in our implementation. 8 frames were sampled in each video, each resized to 256×256 and cropped randomly to 224×224. We obtain the feature vector from the convolutional layers right before the final classification layer, as they encode detailed spatial and temporal features. The Adam optimizer with a learning rate of 10^{-5} was used. We employed the 5-way N-shot accuracies to assess the model's performance. These metrics assess the model's capacity to identify the correct class from the five potential classes, each with a few examples per class. The reported accuracies were derived by calculating the mean accuracy over 10,000 randomly selected episodes

[1] https://pytorch.org/vision/stable/models/video_resnet.html

from the meta-testing set,

We consider two FSL settings to investigate the effect of pre-training on the federated FSL performance. In the first setting, we meta-train 3D-CNN models from scratch with random weights. This setting allows us to understand better the effectiveness of different feature backbones and the meta-learning algorithm. In the second setting, we follow the standard practice [39, 38] in the literature by utilizing the pre-trained feature backbones. Specifically, in our experiments, the weights of 3D-CNN models, which are pre-trained on K400, are provided by Pytorchvideo and Torchvision. Accordingly, for K100, we do not use pre-training for any of the methods since applying K400 for backbone models would violate the FSL assumption.

4.3.2 FL settings

We employed two FL algorithms, FedAvg and FedProx. Our study encompassed using IID and non-IID datasets, allowing for a comprehensive assessment of model performance under varying data distribution scenarios. The experiments were conducted with different numbers of clients, including 2, 4, 8, 16, and 32 clients for training, while we kept one novel client for testing. Each training session extended over a duration of 100 to 200 rounds, with each round consisting of 4 local epochs for every client. For FedProx, we introduced a proximal regularization term with a proxima μ parameter set to 0.001, enriching our exploration of algorithmic behavior in the presence of this regularization. This meticulously designed experimental framework facilitated a thorough evaluation of the algorithms' adaptability and performance across various practical FL scenarios.

Data partitioning. To obtain IID datasets, we evenly distribute all classes among clients. This ensures each client has all classes of the entire dataset. In contrast, for the non-IID setting, we aim to create distinct and specialized data partitions for each client. To achieve this, the number of shared classes between clients is minimal, promoting diversity and individuality in the data each client holds. This strategy enhances the model's ability to handle more complex and diverse real-world scenarios by simulating data distribution variations. Noteworthy, in the non-IID setup, it's imperative to allocate at least 5 distinct classes to each client to support effective 5-way N-shot learning.

4.4. Result Analysis

4.4.1 Performance of different feature backbones

We report results in Tables 1 and 2 for three 3D-CNN backbones (R3D, I3D, and Slow) in different data distribution scenarios (centralized, IID, and non-IID). In these experiments, we use 4 clients for FL settings. Table 1 evaluates models in 1-shot and 5-shot learning tasks on the K100 dataset without pre-training. We can observe that the Slow backbone achieves the highest accuracy in most cases due to its excellent ability to capture semantic information and motion variation in video frames. Moreover, the accuracy of all models decreases when we change the settings from centralized to FL setting. This is because of data heterogeneity and the diversity of human actions across multiple clients, which may hamper the FL performance in K100. Also, for the FL scenario, all models perform better with IID since the non-IID has an unbalanced class distribution.

Table 2 provides insights into the performance of three pre-trained 3D-CNN models on the SSv2 dataset. Again, Slow yields better performances than R3D and I3D in most cases. Interestingly, with pre-training, the accuracy of FL settings is even higher than that of the centralized setting. In addition, the results of IID and non-IID are comparable. The obtained results clearly show the advantages of FL for FSL tasks, especially when combined with pre-trained feature backbones. Pre-training models can help transfer not only the semantic information from the external data to subsequent steps in the training process but also meta-knowledge among clients. Hence, the global model, meta-trained by FL across clients, can adapt to new tasks more effectively. In other words, combining the pre-training with FL can significantly improve the generalization ability of meta-learners to novel clients and alleviate the data heterogeneity issue among FL clients.

Using SSv2, we further investigate the impact of pre-training on few-shot action recognition, where we employ Slow with random weight initialization. We can see that the accuracy in all settings drops significantly. The centralized accuracy is even close to the random guess on 5-way tasks. Our centralized results are consistent with the ones of other meta-learning methods reported in [39]. The poor performance can be attributed to the characteristics of SSv2. Unlike K100, which focuses on spatial appearance, SSv2 requires more complex spatiotemporal reasoning [1]. Hence, it is challenging to learn discriminative representations when meta-training the feature backbone from scratch on this dataset. Surprisingly, using FL settings, we can yield much higher accuracy than that of the centralized counterpart. With FL model aggregation in each round, the meta-knowledge can be transferred among the clients and help the model learn the spatiotemporal features more effectively.

4.4.2 Performance of different FL methods

Tables 3 and 4 present the results of an advanced FedProx [19] in our framework compared to FedAvg [22] on the K100 and SSv2 datasets, respectively. Specifically, we use Slow as the feature backbone and conduct experiments in both IID and non-IID data settings with 4 clients. In general, while FedProx shows inferior performance on the K100 dataset, it brings considerable improvements on the

Table 1. Comparisons of different 3D-CNN backbones on K100.

	Centralized		IID		non-IID	
Models	1 shot	5 shot	1 shot	5 shot	1 shot	5 shot
R3D	46.16%	48.94%	41.33%	46.68%	36.24%	**45.26%**
I3D	37.82%	47.78%	36.13%	44.44%	34.18%	42.12%
Slow	**47.88%**	**59.16%**	**42.22%**	**54.28%**	**36.46%**	44.16%

Table 2. Comparisons of different 3D-CNN backbones on SSv2.

	Centralized		IID		non-IID	
Models	1 shot	5 shot	1 shot	5 shot	1 shot	5 shot
R3D	30.54%	33.48%	38.40%	49.08%	38.52%	49.78%
I3D	41.12%	**60.82%**	43.36%	**61.02%**	42.26%	61.08%
Slow	**44.44%**	60.66%	**45.90%**	60.82%	**45.48%**	**61.74%**
Slow(Random)	23.28%	24.62%	32.90%	45.02%	35.88%	44.4%

Table 3. Comparisons of different FL methods on K100.

	IID		non-IID	
FL algs	1 shot	5 shot	1 shot	5 shot
FedAvg	**42.22%**	**54.28%**	**36.46%**	**44.16%**
FedProx	40.98%	51.00%	35.10%	43.18%

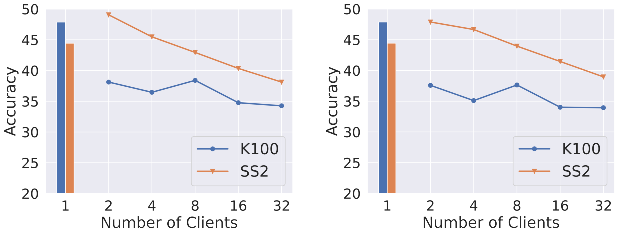
Table 4. Comparisons on different FL methods on SSv2.

	IID		non-IID	
FL algs	1 shot	5 shot	1 shot	5 shot
FedAvg	45.90%	60.82%	45.48%	61.74%
FedProx	**47.24%**	**65.68%**	**46.66%**	**65.68%**

SSv2 dataset compared to FedAvg. This observation indicates the impact of pre-training not only on the performance of 3D-CNN backbones but also on FL algorithms. Moreover, thanks to the proximal term, FedProx shows smaller performance drops in non-IID cases than in IID cases.

4.4.3 Effect of different number of clients

We now investigate the impact of the number of clients in our proposed benchmark under the non-IID setting. By sequentially varying the number of participating clients in data partitioning from 2 to 32, we simultaneously reduce the data amount of each client, simulating cross-device scenarios. The general trend of both FedAvg and FedProx is their performance significantly drops as the number of clients increases. On SSv2 with model pre-training, both FedAvg and FedProx outperform the centralized learning (denoted by 1 client) with 2 and 4 clients. However, running these FL algorithms with more than 4 clients yields inferior performance because the generalization ability of the meta-learner is decreased if training data in each client is insufficient. Looking deeper, FedProx shows its slight effectiveness with less steep declines. These observations raise a need for new FL algorithms for the action recognition task in cross-device scenarios.



(a) FedAvg

(b) FedProx

Figure 2. Result of non-IID federated 1-shot learning on various datasets. The bars indicate the centralized learning accuracy.

4.4.4 Comparison with state-of-the-art on few-shot action recognition

In this section, we compare FedFSLAR with several state-of-the-art methods of centralized few-shot action recognition. Here, we included five best-performing settings of our method for comparison, each using Protonet as the meta-learning algorithm and Slow as the backbone network, which can be either centralized learning or one of the two FL algorithms (FedAvg or FedProx) with 2 and 4 clients. We compare our method with the baseline methods such as Meta-Baseline [39], Baseline Plus [39], and also state-of-the-art (SOTA) methods, including CMN [38], OTAM [4], CMOT [21], TRX [24], HyRSM [31], SloshNet [35], and TADRNet [30]. The results are shown in Table 5.

For a fair comparison on K100, the feature backbones of competitors are initialized randomly, relevant to our first FSL setting without pre-training. Here, the results of corresponding competitors are reported in [39]. We can see that FedFSLAR outperforms the other methods in FSL settings. Centralized learning has the highest accuracy among our variants. Moreover, without pre-training, the spatiotemporal feature backbones, which are meta-trained by either FL or centralized learning, are proven more effective than

Table 5. Comparison with SOTA Few-Shot Action Recognition Methods.

Method	Setting	K100		SSv2	
		1 shot	5 shot	1 shot	5 shot
Meta-Baseline [39]	Centralized Learning	42.46%	49.78%	33.6%	43.0%
Baseline Plus [39]	Centralized Learning	46.24%	56.92%	46.04%	61.10%
CMN [38]	Centralized Learning	40.37%	50.27%	34.4%	43.8%
OTAM [4]	Centralized Learning	44.37%	50.07%	42.8%	52.3%
CMOT[21]	Centralized Learning	-	-	46.8%	55.9%
TRX [24]	Centralized Learning	-	-	42.0%	64.6%
HyRSM [31]	Centralized Learning	-	-	**54.3%**	**69.0%**
SloshNet [35]	Centralized Learning	-	-	46.5%	68.3%
TADRNet [30]	Centralized Learning	-	-	43%	61.1%
FedFSLAR	Centralized Learning	**47.88%**	**59.16%**	44.44%	60.66%
FedFSLAR	FedAvg(2 clients, IID)	42.86%	57.42%	48.92%	66.02%
FedFSLAR	FedProx(2 clients, IID)	42.26%	56.12%	48.06%	66.06%
FedFSLAR	FedAvg(4 clients, IID)	42.22%	54.28%	45.90%	60.82%
FedFSLAR	FedProx(4 clients, IID)	40.98%	51%	47.24%	65.68%

Note: Best and second-best results are denoted in bold and underlined, respectively.

Meta-Baseline, Baseline Plus, CMN, and OTAM, using frame-level features. 3D-CNN models used in our work can generate more discriminative representation, hence better generalizing to new tasks.

For SSv2, we employ pre-trained Slow for evaluation. Note that, among our competitors, CMOT adopted the spatiotemporal backbone network (C3D) pre-trained on Sport-1M, while other SOTA methods used the frame-level feature backbones pre-trained on ImageNet. With SSv2, We can observe that our FedFSLAR outperforms other methods except for HyRSM and SloshNet (for a 5-shot setting). When using FL with two clients, our method performs better than its spatiotemporal counterpart (i.e., CMOT) by a large margin, while we yield a competitive performance with recent SOTA methods like TRX, HyRSM, SloshNet, and TADRNet. Notably, our FedFSLAR has a simpler pipeline than these methods, often involving complex attention mechanisms or task-adaptive modules. Interestingly, some of our FL variants achieve higher accuracy than the SOTA centralized methods. This suggests that FL can help improve the generalization ability of the model by aggregating meta-knowledge from multiple clients.

5. Conclusion and Future Work

In this paper, we introduce a novel framework, FedFSLAR, for studying the problem of few-shot video-based action recognition under the FL settings. The proposed framework comprises 3D-CNN models as feature backbones for learning robust representation and a meta-learning algorithm, i.e., ProtoNet [25], which allows us to recognize unseen actions with a few video samples. FL algorithms, i.e., FedAvg [22], and FedProx [19], are employed to enable privacy-preserving decentralized learning. Especially, our framework provides the flexibility to adapt various advanced models and algorithms for each component. Moreover, extensive experimental results on standard datasets under practical FL settings verify the effectiveness of FedFSLAR for solving the recognition task. Also, a range of valuable insights are drawn and presented to promote this research problem.

In the future, FedFSLAR will be straightforwardly extended with more diverse spatiotemporal models, meta-learning algorithms, and FL algorithms. Besides the current video-based approach, skeleton-based action recognition will also be explored in the FL scenario. To exploit the massive collection of unlabeled data from heterogeneous user devices, we plan to develop a self-supervised pre-training method under the FL scenario to train feature backbone networks to extract helpful video representation. Furthermore, in addition to the issue of limited training data, the challenging domain shift issue should also be investigated due to its large significance.

Acknowledgments

This research was funded by Nazarbayev University under Faculty Development Competitive Research Grant Program (No. 11022021FD2925).

References

[1] Gedas Bertasius, Heng Wang, and Lorenzo Torresani. Is space-time attention all you need for video understanding? In *ICML*, volume 2, page 4, 2021. 6

[2] Mina Bishay, Georgios Zoumpourlis, and Ioannis Patras. Tarn: Temporal attentive relation network for few-shot and zero-shot action recognition. *arXiv preprint arXiv:1907.09021*, 2019. 3

[3] Dongqi Cai, Shangguang Wang, Yaozong Wu, Felix Xiaozhu Lin, and Mengwei Xu. Federated few-shot learning for mobile nlp. In *Proceedings of the 29th Annual International Conference on Mobile Computing and Networking*, pages 1–17, 2023. 3

[4] Kaidi Cao, Jingwei Ji, Zhangjie Cao, Chien-Yi Chang, and Juan Carlos Niebles. Few-shot video classification via temporal alignment. In *CVPR*, pages 10615–10624, 2020. 3, 7, 8

[5] João Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *CVPR*, pages 4724–4733, 2017. 1, 2, 5

[6] Jiao Chen, Jianhua Tang, and Weihua Li. Industrial edge intelligence: Federated-meta learning framework for few-shot fault diagnosis. *IEEE Transactions on Network Science and Engineering*, 2023. 3

[7] Keval Doshi and Yasin Yilmaz. Federated learning-based driver activity recognition for edge devices. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3338–3346, 2022. 3

[8] Chenyou Fan and Jianwei Huang. Federated few-shot learning with adversarial learning. In *2021 19th International Symposium on Modeling and Optimization in Mobile, Ad hoc, and Wireless Networks (WiOpt)*, pages 1–8. IEEE, 2021. 2, 3

[9] Haoqi Fan, Tullie Murrell, Heng Wang, Kalyan Vasudev Alwala, Yanghao Li, Yilei Li, Bo Xiong, Nikhila Ravi, Meng Li, Haichuan Yang, et al. Pytorchvideo: A deep learning library for video understanding. In *Proceedings of the 29th ACM international conference on multimedia*, pages 3783–3786, 2021. 5

[10] Christoph Feichtenhofer. X3d: Expanding architectures for efficient video recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 203–213, 2020. 2

[11] Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. Slowfast networks for video recognition. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 6202–6211, 2019. 5

[12] Yuqian Fu, Li Zhang, Junke Wang, Yanwei Fu, and Yu-Gang Jiang. Depth guided adaptive meta-fusion network for few-shot video recognition. In *Proceedings of the 28th ACM International Conference on Multimedia*, pages 1142–1151, 2020. 3

[13] Raghav Goyal, Samira Ebrahimi Kahou, Vincent Michalski, Joanna Materzynska, Susanne Westphal, Heuna Kim, Valentin Haenel, Ingo Fruend, Peter Yianilos, Moritz Mueller-Freitag, et al. The" something something" video database for learning and evaluating visual common sense. In *ICCV*, pages 5842–5850, 2017. 5

[14] Chaoyang He, Murali Annavaram, and Salman Avestimehr. Group knowledge transfer: Federated learning of large cnns at the edge. *Advances in Neural Information Processing Systems*, 33:14068–14080, 2020. 3

[15] Ngan Dao Hoang, Dat Tran-Anh, Manh Luong, Cong Tran, and Cuong Pham. Federated few-shot learning for cough classification with edge devices. *Applied Intelligence*, pages 1–13, 2023. 3

[16] Peter Kairouz, H Brendan McMahan, Brendan Avent, Aurélien Bellet, Mehdi Bennis, Arjun Nitin Bhagoji, Kallista Bonawitz, Zachary Charles, Graham Cormode, Rachel Cummings, et al. Advances and open problems in federated learning. *Foundations and Trends® in Machine Learning*, 14(1–2):1–210, 2021. 1

[17] Yu Kong and Yun Fu. Human action recognition and prediction: A survey. *International Journal of Computer Vision*, 130(5):1366–1401, 2022. 1

[18] Sai Kumar Dwivedi, Vikram Gupta, Rahul Mitra, Shuaib Ahmed, and Arjun Jain. Protogan: Towards few shot learning for action recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, pages 0–0, 2019. 3

[19] Tian Li, Anit Kumar Sahu, Manzil Zaheer, Maziar Sanjabi, Ameet Talwalkar, and Virginia Smith. Federated optimization in heterogeneous networks. *Proceedings of Machine learning and systems*, 2:429–450, 2020. 1, 2, 5, 6, 8

[20] Ze Liu, Jia Ning, Yue Cao, Yixuan Wei, Zheng Zhang, Stephen Lin, and Han Hu. Video swin transformer. In *CVPR*, pages 3202–3211, 2022. 5

[21] Su Lu, Han-Jia Ye, and De-Chuan Zhan. Few-shot action recognition with compromised metric via optimal transport. *arXiv preprint arXiv:2104.03737*, 2021. 3, 7, 8

[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. 2, 3, 4, 6, 8

[23] Xiaomin Ouyang, Zhiyuan Xie, Jiayu Zhou, Guoliang Xing, and Jianwei Huang. Clusterfl: A clustering-based federated learning system for human activity recognition. *ACM Transactions on Sensor Networks*, 19(1):1–32, 2022. 3

[24] Toby Perrett, Alessandro Masullo, Tilo Burghardt, Majid Mirmehdi, and Dima Damen. Temporal-relational crosstransformers for few-shot action recognition. In *CVPR*, pages 475–484, 2021. 3, 7, 8

[25] Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. In *NIPS*, page 4080–4090, 2017. 2, 4, 8

[26] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. Learning spatiotemporal features with 3d convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, pages 4489–4497, 2015. 3

[27] Du Tran, Heng Wang, Lorenzo Torresani, Jamie Ray, Yann LeCun, and Manohar Paluri. A closer look at spatiotemporal convolutions for action recognition. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6450–6459, 2018. 3, 5

[28] Limin Wang, Yuanjun Xiong, Zhe Wang, Yu Qiao, Dahua Lin, Xiaoou Tang, and Luc Van Gool. Temporal segment networks for action recognition in videos. *IEEE transactions on pattern analysis and machine intelligence*, 41(11):2740–2755, 2018. 5

[29] Song Wang, Xingbo Fu, Kaize Ding, Chen Chen, Huiyuan Chen, and Jundong Li. Federated few-shot learning. In

Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, KDD '23, page 2374–2385, New York, NY, USA, 2023. Association for Computing Machinery. 3

[30] Xiao Wang, Weirong Ye, Zhongang Qi, Guangge Wang, Jianping Wu, Ying Shan, Xiaohu Qie, and Hanzi Wang. Task-aware dual-representation network for few-shot action recognition. *IEEE Transactions on Circuits and Systems for Video Technology*, 2023. 7, 8

[31] Xiang Wang, Shiwei Zhang, Zhiwu Qing, Mingqian Tang, Zhengrong Zuo, Changxin Gao, Rong Jin, and Nong Sang. Hybrid relation guided set matching for few-shot action recognition. In *CVPR*, pages 19916–19925, 2022. 3, 7, 8

[32] Yaqing Wang, Quanming Yao, James T Kwok, and Lionel M Ni. Generalizing from a few examples: A survey on few-shot learning. *ACM computing surveys (csur)*, 53(3):1–34, 2020. 2

[33] Yongqin Xian, Bruno Korbar, Matthijs Douze, Lorenzo Torresani, Bernt Schiele, and Zeynep Akata. Generalized few-shot video classification with video retrieval and feature generation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(12):8949–8961, 2021. 3

[34] Zhiwen Xiao, Xin Xu, Huanlai Xing, Fuhong Song, Xinhan Wang, and Bowen Zhao. A federated learning system with enhanced feature extraction for human activity recognition. *Knowledge-Based Systems*, 229:107338, 2021. 3

[35] Jiazheng Xing, Mengmeng Wang, Yong Liu, and Boyu Mu. Revisiting the spatial and temporal modeling for few-shot action recognition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 3001–3009, 2023. 7, 8

[36] Guangle Yao, Tao Lei, and Jiandan Zhong. A review of convolutional-neural-network-based action recognition. *Pattern Recognition Letters*, 118:14–22, 2019. 1

[37] Bin Zhang, Jingya Wang, Junyi Fu, and Jinxiang Xia. Driver action recognition using federated learning. In *Proceedings of the 7th International Conference on Communication and Information Processing*, pages 74–77, 2021. 3

[38] Linchao Zhu and Yi Yang. Compound memory networks for few-shot video classification. In *ECCV*, September 2018. 2, 3, 5, 6, 7, 8

[39] Zhenxi Zhu, Limin Wang, Sheng Guo, and Gangshan Wu. A closer look at few-shot video classification: A new baseline and benchmark. *arXiv preprint arXiv:2110.12358*, 2021. 2, 6, 7, 8