

CurFed-CHARM: Curriculum Sequential Federated Learning with a Channel-Hierarchical Action Recognition Model for non-IID Heterogeneous Data in Neuromorphic/Event Cameras

Muhammad Hamza Zafar^a, Syed Kumayl Raza Moosavi^a and Filippo Sanfilippo^{a,*}

^aDepartment of Engineering Sciences, University of Agder, Grimstad, 4879, Norway

ARTICLE INFO

Keywords:

Human Action Recognition
Neuromorphic Vision
Event-Based Cameras
Deep Learning
Human-Robot Interaction (HRI)

ABSTRACT

Human action recognition is a key technology in healthcare monitoring, security surveillance, human-computer interaction, and human-robot interaction. Traditional vision-based approaches, relying on RGB cameras, suffer from motion blur, high bandwidth demands, and poor performance in challenging lighting conditions. Neuromorphic event cameras offer promising solutions but still face challenges related to privacy-preserving model training and decentralized learning for non-IID heterogeneous data. To address these limitations, we propose Curriculum Federated Learning based Channel Hierarchical Action Recognition (*CurFed-CHARM*), a novel framework for neuromorphic vision-based action recognition. *CurFed-CHARM* leverages event-driven sensing and progressive knowledge distillation to enhance real-time recognition accuracy while preserving data privacy across distributed edge devices. Our approach implements a sequential federated learning strategy with attention-enhanced 3D CNNs that process spatio-temporal features from action sequences. The framework begins with a comprehensive client possessing all action classes and progressively integrates additional clients once performance thresholds are achieved. We introduce specialized metrics including Client Convergence Rate, Knowledge Transfer Efficiency, Catastrophic Forgetting Metric, and Communication Efficiency to evaluate the system's performance. Our experimental results demonstrate that *CurFed-CHARM* achieves superior performance with 94.87% average accuracy across all clients on E-FedAct dataset, outperforming traditional federated approaches like FedAvg (92.99%), FedProx (92.42%), and FedCurv (92.09%). The ablation study confirms the critical importance of our Global Temporal Attention mechanism, which significantly enhances knowledge transfer efficiency by 60% compared to models without this component. Additionally, *CurFed-CHARM* reduces communication costs by 27.3% compared to conventional federated learning methods through its strategic client integration strategy.

1. Introduction

Human action recognition has emerged as a critical technology with applications in healthcare monitoring, security surveillance, human-computer interaction, human-robot interaction, and assistive technologies [1]. Accurate recognition of human movements enables intelligent systems to adapt to user behaviours, detect anomalies, and provide real-time assistance in dynamic environments. As smart environments continue to expand, the demand for robust and efficient action recognition systems has grown significantly.

Traditional action recognition methods primarily rely on RGB cameras [2], which, despite their widespread use, suffer from inherent limitations. These systems are prone to motion blur, struggle in low-light conditions, and require high bandwidth for frame-based data processing. Neuromorphic vision sensors, also known as event cameras, offer a promising alternative by capturing visual information through an event-driven approach [3]. Unlike conventional cameras, event cameras respond to changes in the scene rather than recording redundant frames, enabling low-latency processing and energy-efficient operation. However, effectively leveraging event-based data for action recognition remains a challenge

due to the sparse and asynchronous nature of the captured events.

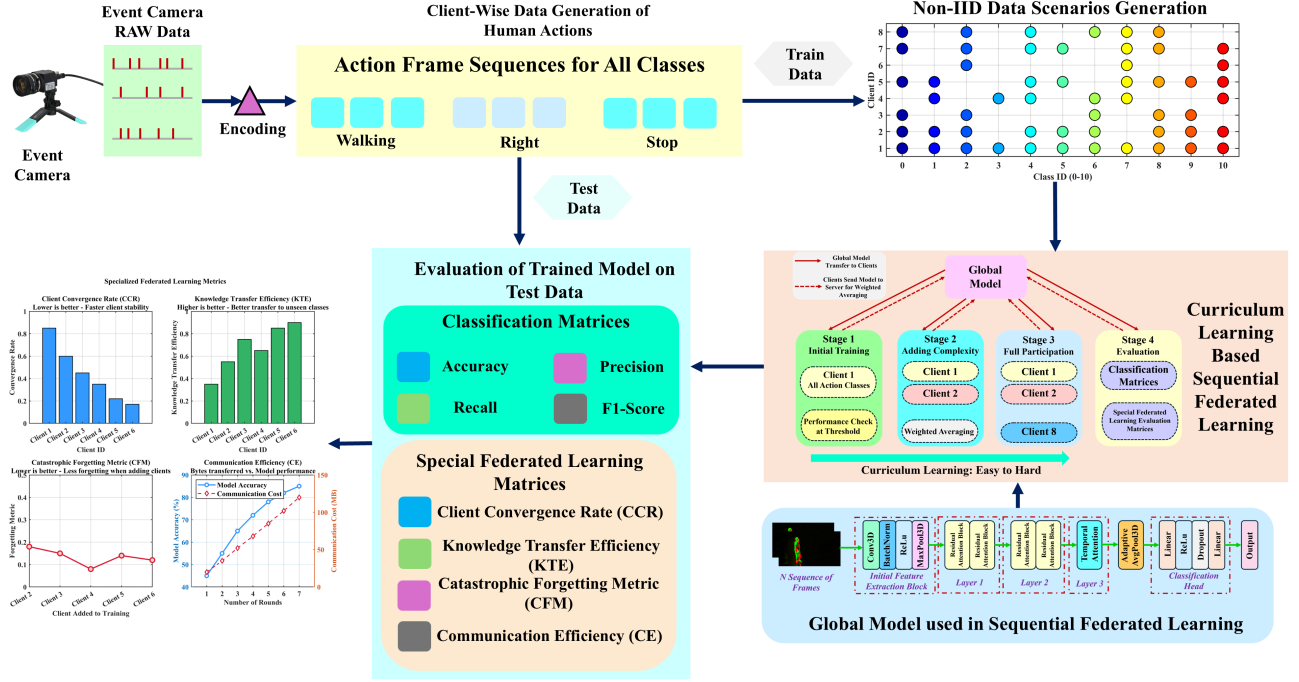
Deep learning models have demonstrated strong performance in action recognition tasks by extracting complex spatio-temporal patterns from data [4]. However, their success typically depends on large-scale labelled datasets and centralized training, which pose significant challenges in privacy-sensitive and resource-constrained settings. In real-world scenarios, data is often distributed across multiple devices, each capturing unique but non-independent and identically distributed (non-IID) data streams. The presence of heterogeneous data across different environments further complicates centralized learning approaches, limiting their generalization capabilities.

Federated learning presents a compelling solution to decentralized model training by enabling multiple devices to collaboratively train a global model without sharing raw data [5]. However, conventional federated learning methods struggle with non-IID data distributions, leading to slow convergence and suboptimal performance. The need for an effective mechanism to handle data heterogeneity while ensuring privacy-preserving training is crucial for advancing neuromorphic action recognition.

To overcome these issues, we introduce *CurFed-CHARM*, a novel Curriculum Sequential Federated Learning framework with a Channel-Hierarchical Action Recognition Model.

*Corresponding author

 muhammad.h.zafar@uia.no (M.H. Zafar); syed.k.moosavi@uia.no (S.K.R. Moosavi); filippo.sanfilippo@uia.no (F. Sanfilippo)
ORCID(s):


 Figure 1: Description of Proposed *CurFed-CHARM*.

Our approach leverages curriculum learning to structure training in a progressive manner, enabling improved generalization across diverse, non-IID datasets. By incorporating federated learning, *CurFed-CHARM* ensures privacy-preserving collaborative training across distributed neuromorphic sensors while mitigating the effects of data heterogeneity. The hierarchical architecture further enhances action recognition by capturing multi-channel spatiotemporal dependencies in event streams.

1.1. Literature Review

Activity recognition has evolved significantly from spatial classification tasks to sophisticated spatio-temporal learning approaches. Early methods relied on handcrafted features in preprocessing stages [6], [7], which limited their adaptability to diverse scenarios. A major advancement came with the formulation of localized action recognition, where temporal bodies of actions needed to be precisely measured and localized in each frame, producing spatio-temporal tubes or action gestures [8]. However, these approaches faced challenges in reducing reaction time, prompting researchers to develop event aggregation techniques. For instance, Innocenti et al. [9] proposed binary representations by evaluating event presence at each pixel within fixed time intervals, followed by binary-to-decimal conversion for frame generation. The quest for reduced inference time and energy efficiency led to the widespread adoption of event cameras for gesture and action recognition tasks. A significant milestone was achieved by Layher et al. [10], demonstrating real-time action recognition at approximately

1,000 frames per second while consuming only 70 mW of power. Building upon this foundation, Baby et al. [6] combined Dynamic Vision Sensors (DVS) with Motion Maps and Motion Boundary Histograms (MBH), outperforming traditional RGB camera methods in both accuracy and processing time. This work was further enhanced by Amir et al. [11], who integrated DVS with TrueNorth neurosynaptic processors, achieving 96.5% accuracy in hand gesture recognition at less than 200mW power consumption.

The integration of event cameras with deep learning models marked another evolutionary leap. Pradhan et al. [12] introduced event memory surfaces to adapt sparse and asynchronous event data for CNN-based models, achieving 94.3% accuracy on real-world datasets. However, CNN-based approaches struggled with efficiently encoding spatiotemporal information from event streams. To address this limitation, Liu et al. [13] proposed the Aligned Compressed Event Tensor (ACE) and Branched Event Net (BET), which efficiently processed sparse, asynchronous event data and achieved state-of-the-art inference speeds while maintaining high accuracy. Recognizing that traditional neural networks were not optimally suited for asynchronous event data, recent research has shifted toward Spiking Neural Networks (SNNs). Liu et al. [14] employed a hierarchical approach that progressively integrated local motion details into global contextual understanding, better matching the event-driven nature of DVS data. Yang et al. [15] further refined this approach with a five-layer convolutional Leaky Integrate-and-Fire (LIF) SNN, incorporating background activity filtering for noise reduction and achieving superior accuracy

on multiple benchmark datasets including DVS-Gesture and IITM-DVS. Most recently, advanced architectures like Spike-HAR++ [16] have introduced parallel spiking transformers optimized for energy-efficient HAR, representing the current state-of-the-art in this domain.

Beyond general activity recognition, event cameras have been successfully applied to specialized tasks. For human fall detection, Chen et al.[17] leveraged a Temporal-Spatial Attention Weighted (TSAW) network achieving robust fall localization under varying lighting conditions, while Prasad et al.[18] introduced the DVSFall dataset with a hybrid 3D-CNN and SNN framework for privacy-preserving detection. Additionally, event cameras have shown promise in egocentric action recognition from wearables [19] and fine-grained tasks such as facial micro-expression recognition [4]. The field continues to benefit from large-scale datasets like HARDVS [20] and N-EPIC-Kitchens [21], which provide valuable benchmarking resources for advancing event-based human activity recognition.

1.2. Motivation

The convergence of event-based vision and federated learning represents a timely and critical advancement in human action recognition systems. Despite significant progress in computer vision, existing approaches continue to face fundamental challenges in real-world deployment—including privacy concerns, bandwidth limitations, power consumption constraints, and performance degradation in adverse conditions. Our work addresses these multifaceted challenges through an innovative integration of neuromorphic sensing with privacy-preserving distributed learning frameworks. By leveraging the inherent advantages of event cameras—microsecond temporal resolution, high dynamic range, and minimal data footprint—alongside curriculum-based federated learning strategies specifically designed for Non-IID data distributions, we enable robust action recognition capabilities that can function reliably in previously prohibitive environments. This approach not only overcomes the technical limitations of conventional RGB-based systems but also aligns with growing societal demands for privacy-conscious technologies and efficient edge computing solutions. The practical implications span critical domains from healthcare monitoring for aging populations to smart manufacturing environments and assistive technologies, where reliable human action understanding must coexist with stringent privacy and resource constraints. To our best knowledge this is first time curriculum federated learning framework is presented for the non-IID heterogeneous data for human actions recognition using event camera.

1.3. Contributions

The contributions of this work are:

- A novel fusion of event camera data with curriculum-based federated learning for human action recognition with Progressive client incorporation strategy that mitigates catastrophic forgetting with Non-IID data

and Optimized knowledge transfer across heterogeneous client distributions.

- Specialized neural architecture with dual spatio-temporal attention for event data processing which includes Channel and temporal attention mechanisms capturing sparse, temporal actions sequences and Efficient feature extraction optimized for edge deployment with event camera data.
- Specialized metrics quantifying client convergence, knowledge transfer, and catastrophic forgetting are used which provides evaluation tools that provide deeper insights beyond conventional accuracy measurements and framework for analyzing cross-client learning dynamics in distributed event-based systems.
- Threshold-based client incorporation approach ensuring model stability with non-IID heterogeneous data with Adaptive sequencing strategy addressing variable class distributions across federated clients which reduces communication overhead while maximizing performance with diverse event data.

2. Problem Formulation

2.1. Description of non-IID data setup

In our work, we address the challenging scenario of federated learning with non-Independent and Identically Distributed (non-IID) data across clients in the context of video-based action recognition. Unlike conventional federated learning setups that assume relatively uniform class distributions across clients, we deliberately construct a heterogeneous environment where each client possesses a distinct subset of action classes. This reflects real-world scenarios where edge devices capture different human activities based on their deployment context (e.g., surveillance cameras in different locations capturing location-specific activities). Specifically, we designate Client 1 as the anchor client that contains examples from all action classes $C = \{c_1, c_2, \dots, c_K\}$, where K represents the total number of action classes. For each remaining client $i \in \{2, 3, \dots, N\}$, we assign a random subset of classes $C_i \subset C$, where $|C_i| \geq m$ and m represents a minimum number of classes per client. This creates a challenging learning environment where the global model must integrate knowledge from distributed, partially overlapping datasets without direct access to the complete set of examples for each class at each client. The degree of non-IIDness can be quantified using the Jaccard similarity coefficient between class distributions across clients:

$$J(C_i, C_j) = \frac{|C_i \cap C_j|}{|C_i \cup C_j|} \quad (1)$$

2.2. Mathematical Formulation of Sequential Federated Learning (SFL)

Sequential Federated Learning (SFL) represents a novel paradigm where clients participate in the federated training

process incrementally rather than simultaneously. We formalize SFL as an iterative process over R global rounds. At each round $r \in \{1, 2, \dots, R\}$, only a subset of clients $S_r \subseteq \{1, 2, \dots, N\}$ actively participate in the training process. The global model parameters at round r are denoted as $\mathbf{w}^r$. For each active client $i \in S_r$, the local optimization problem is defined as:

$$\mathbf{w}_i^r = \arg \min_{\mathbf{w}} \frac{1}{|D_i|} \sum_{(x,y) \in D_i} \mathcal{L}(\mathbf{w}, x, y) \quad (2)$$

where D_i represents the local dataset of client i , (x, y) denotes a sample-label pair, and $\mathcal{L}$ is the loss function (cross-entropy loss in our action recognition task). Each client initializes its local model with the global parameters $\mathbf{w}^r$ and performs E epochs of local training to obtain $\mathbf{w}_i^r$. The global model is then updated through a weighted aggregation of the local models:

$$\mathbf{w}^{r+1} = \sum_{i \in S_r} \alpha_i \mathbf{w}_i^r \quad (3)$$

where α_i represents the weight assigned to client i , calculated based on the class diversity of the client: $\alpha_i = \frac{|C_i|}{\sum_{j \in S_r} |C_j|}$. This weighting scheme gives higher importance to clients with greater class diversity, thus promoting knowledge transfer across the action recognition task space.

2.3. Mathematical Formulation of Curriculum Learning in SFL

We integrate curriculum learning principles into our SFL framework by carefully controlling the sequence in which clients are introduced to the global training process. This approach is motivated by the observation that learning from simpler, more generalized data distributions before tackling complex, specialized ones can lead to more robust models. For a given performance threshold τ , we determine client participation at round $r + 1$ based on the global model's performance at round r :

$$S_{r+1} = \begin{cases} S_r \cup \{j\}, & \text{if } A(\mathbf{w}^r, D_{S_r}) \geq \tau \\ & \text{and } j = \min(\{1, 2, \dots, N\} \setminus S_r) \\ S_r, & \text{otherwise} \end{cases} \quad (4)$$

where $A(\mathbf{w}^r, D_{S_r})$ represents the average accuracy of the global model $\mathbf{w}^r$ across the validation datasets of all currently active clients S_r . We initialize $S_1 = \{1\}$, beginning with the anchor client that contains all action classes. This formulation ensures that new clients are incorporated only when the model has achieved sufficient performance on the current active client set, creating a natural curriculum that progresses from general to specific data distributions.

2.4. Strategy for gradual exposure of action classes in SFL

Our approach implements a strategic gradual exposure of action classes throughout the federated learning process,

capitalizing on the class heterogeneity across clients. The initial phase focuses exclusively on Client 1, which contains all K action classes, allowing the model to develop a general understanding of the entire action space. As new clients are sequentially integrated according to our curriculum learning strategy, the model encounters specialized distributions of these classes, providing refined examples that help discriminate subtle variations within each action category. This gradual exposure can be quantified through the cumulative class exposure function $E(r)$ at round r :

$$E(r) = \left| \bigcup_{i \in S_r} C_i \right| \quad (5)$$

Importantly, since Client 1 contains all classes, $E(1) = K$, ensuring that the model encounters all classes from the beginning but receives increasingly diverse manifestations of these classes as training progresses. To further structure this exposure, we assign class subsets to clients such that classes with similar temporal dynamics or visual characteristics appear together more frequently, creating natural groupings that facilitate hierarchical learning. This approach mirrors how humans learn to recognize actions—first distinguishing broad categories (e.g., stationary vs. moving actions) before refining perception to distinguish subtle variations within similar action types. Through this deliberate orchestration of class exposure, our SFL framework encourages the model to develop robust feature representations that generalize effectively across the diverse data distributions encountered in real-world action recognition scenarios.

3. Dataset Descriptions and Experimental Design

This section describes the two datasets used to test our proposed model. The first is E-FedAct, which we collected ourselves. The second is the PAF dataset, a well-known dataset for human action recognition. We chose the PAF dataset because it works well for developing federated learning techniques for human action recognition with event cameras. This section also explains how we simulated non-IID scenarios for both datasets. We describe how classes were randomly distributed among clients to create a realistic non-IID setup.

3.1. E-FedAct Dataset:

We utilized the Prophesee EVK4 neuromorphic vision sensor (1280×720 pixel resolution) to create a comprehensive action recognition dataset i.e. E-FedAct. We allocated eleven distinct human actions (coded 0-10):

To ensure robust data diversity, eight participants each performed every action a minimum of ten repetitions. The action selection was deliberate, encompassing fundamental locomotion, directional navigation, and postural transitions.

Environmental variability was introduced by conducting recordings under both optimal and suboptimal lighting conditions. This approach specifically tests the neuromorphic

Table 1

Action categories in the E-FedAct dataset

Action	Label
Bend Down	0
Getting Up	1
Jumping	2
Left	3
Move Back	4
Move Forward Fast	5
Move Forward Slow	6
Right	7
Stop	8
Turn Around	9
Walking	10

Table 2

Action categories in the PAF dataset

Action	Label
Arm Crossing	0
Getting Up	1
Jumping	2
Kicking	3
Picking Up	4
Sitting Down	5
Throwing	6
Waving	7
Turning Around	8
Walking	9

sensor’s capability to detect motion across different illumination scenarios. Unlike conventional frame-based cameras that capture images at predetermined intervals, the Prophesee EVK4 operates on an event-driven principle, registering only brightness changes at the pixel level as they occur in real-time. This property makes such sensors particularly advantageous for motion analysis applications.

3.1.1. Technical Specifications of Event Data

The neuromorphic camera generates an asynchronous stream where each event is characterized by a four-element tuple:

$$e = (x, y, t, p) \quad (6)$$

In this representation $(x, y) \in \mathbb{R}^2$ indicates the spatial coordinates within the sensor’s resolution, $t \in \mathbb{R}^+$ represents the timestamp with microsecond precision and $p \in \{-1, +1\}$ denotes polarity reflecting the direction of brightness change

An event is triggered when logarithmic intensity changes exceed a predetermined threshold θ :

$$\log \left(\frac{I(x, y, t)}{I(x, y, t_{\text{last}})} \right) = \pm \theta \quad (7)$$

Where $I(x, y, t)$ signifies intensity at position (x, y) and time t , while t_{last} refers to the previous event timestamp at that location.

3.1.2. Data Processing Framework

To transform the asynchronous event stream into structured representations compatible with conventional analysis techniques, we accumulated events across defined temporal windows. For any interval $[t, t + \Delta t]$, we computed frame representation $F(x, y)$ as:

$$F(x, y) = \sum_{e_i \in \mathcal{E}_{t, \Delta t}(x, y)} p_i \quad (8)$$

Where $\mathcal{E}_{t, \Delta t}(x, y)$ encompasses all events at position (x, y) within the specified time window, and p_i represents the polarity of each individual event.

The resulting representation $F(x, y)$ captures net brightness fluctuations during the accumulation period. For human action recognition applications, the accumulation time used for event to frame conversion is 30ms.

3.2. PAF Dataset:

The action recognition dataset comprises event-based recordings captured using the DAVIS346 neuromorphic vision sensor, featuring 15 subjects performing 10 distinct actions—arm-crossing, getting-up, kicking, picking-up, jumping, sitting-down, throwing, turning-around, walking, and waving—in an empty office environment. The labels of the actions are presented in Table. 2. Each subject performs each action three times, with recordings lasting approximately five seconds each, captured from three different camera positions and distances to ensure variability. The dataset includes a total of 450 recordings at a resolution of 346×260 pixels, recorded as asynchronous event streams in .aedit format [22].

3.3. Simulation of non-IID Scenario

We designed a realistic non-IID scenario that mirrors the challenges of federated learning in real-world situations. Instead of simply dividing a uniform dataset, we created different class distributions across clients by giving each client access to only certain action classes. Client 1 acts as an anchor and has access to all 11 action classes for E-FedAct and all 10 classes for the PAF dataset. The remaining clients (2-8 for E-FedAct and 2-15 for PAF) each receive a random subset containing 3-10 classes. During dataset construction, our data loading system filtered the action sequences so each client only received samples from their assigned classes. This created distinct data distributions with only partial overlap between clients. This setup mimics real-world scenarios like traffic monitoring or surveillance networks, where each device naturally captures only certain types of actions based on its location and context. The result is a challenging environment where the global model must learn about all action classes even though no single client (except the anchor) sees examples of every action type.

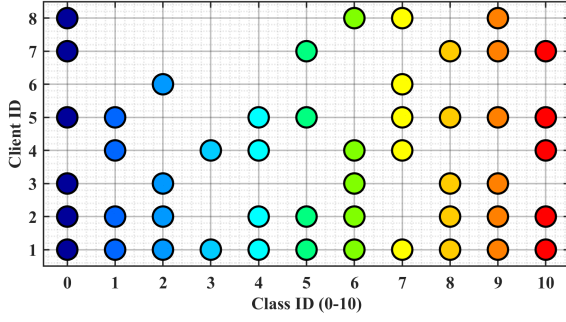


Figure 2: Graphical Description of non-IID Data setup for E-FedAct Dataset.

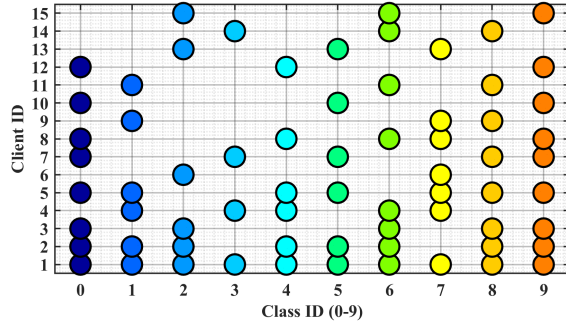


Figure 3: Graphical Description of non-IID Data setup for PAF Dataset.

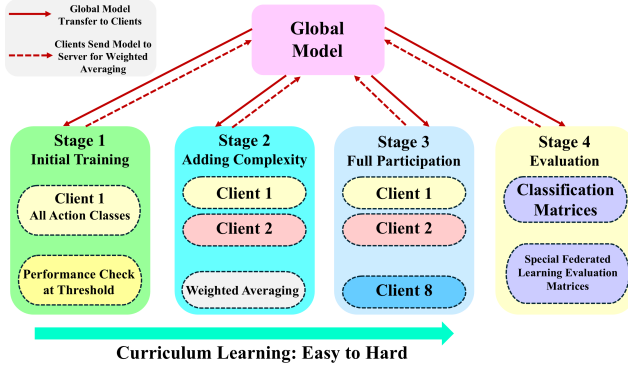


Figure 4: Description of Curriculum Learning Based Sequential Federated Learning for non-IID Data of Human Actions.

4. Proposed Curriculum Learning Approach

4.1. Curriculum design for progressive learning in FL

Our proposed curriculum learning approach for federated learning introduces a novel progressive training strategy that systematically integrates heterogeneous clients into the training process. Unlike conventional federated learning where all clients participate simultaneously, we implement a carefully designed curriculum that controls client participation based on model performance and data characteristics. The curriculum starts with Client 1 (our anchor client) possessing a complete set of action classes $C = c_1, c_2, \dots, c_K$, then progressively incorporates additional clients based on

performance thresholds. Formally, we define a curriculum function $\Phi_r : \mathbb{R}^d \times \mathcal{P}(1, 2, \dots, N) \rightarrow \mathcal{P}(1, 2, \dots, N)$ that determines the set of active clients at round $r+1$ based on the current model parameters $\mathbf{w}^r$ and the currently active client set S_r :

$$S_{r+1} = \Phi_r(\mathbf{w}^r, S_r) \quad (9)$$

Our implementation uses a performance-triggered curriculum where:

$$\Phi_r(\mathbf{w}^r, S_r) = \begin{cases} S_r \cup \{j_r\}, & \text{if } A(\mathbf{w}^r, D(S_r)) \geq \tau_r, \\ S_r, & \text{otherwise.} \end{cases} \quad (10)$$

where $A(\mathbf{w}^r, D_{S_r})$ represents the average test accuracy across all currently active clients, τ_r is the performance threshold at round r , and $j_r = \min(1, 2, \dots, N \setminus S_r)$ is the next client to be added. This formulation ensures that new clients are incorporated only when the model has achieved sufficient performance on the current set. To enhance this curriculum, we further define a client difficulty score δ_i for each client i based on three factors: class uniqueness, data distribution characteristics, and the estimated domain gap from previously seen clients:

$$\delta_i = \lambda_1 \cdot \frac{|C_i \setminus \bigcup_{j \in S_r} C_j|}{|C_i|} + \lambda_2 \cdot H(p_i) + \lambda_3 \cdot D(p_i, p_{S_r}) \quad (11)$$

where $H(p_i)$ represents the entropy of the class distribution in client i , $D(p_i, p_{S_r})$ measures the distribution divergence (e.g., Wasserstein distance) between client i and the aggregate distribution of active clients, and $\lambda_1, \lambda_2, \lambda_3$ are weighting hyperparameters. This difficulty score allows us to refine the client selection process for more advanced curriculum designs.

4.2. Integration of CL with FL optimization (e.g., adaptive client weighting)

We integrate curriculum learning with federated optimization through an adaptive weighting mechanism that evolves throughout the training process. Our approach extends beyond simple data-volume-based weighting by incorporating curriculum-aware aggregation that assigns importance to clients based on their curriculum stage and contribution to the overall learning objective. At each round r , the global model update follows:

$$\mathbf{w}^{r+1} = \mathbf{w}^r - \eta_r \sum_{i \in S_r} \beta_i^r \nabla F_i(\mathbf{w}^r) \quad (12)$$

where η_r is the global learning rate, $\nabla F_i(\mathbf{w}^r)$ represents the gradient computed by client i , and β_i^r is the curriculum-adaptive weight assigned to client i at round r . We define this adaptive weight as:

$$\beta_i^r = \frac{\alpha_i \cdot \gamma_i^r}{\sum_{j \in S_r} \alpha_j \cdot \gamma_j^r} \quad (13)$$

Here, $\alpha_i = \frac{|C_i|}{\sum_{j \in S_r} |C_j|}$ represents the class diversity weight, and γ_i^r is a curriculum-specific coefficient that adjusts the

client's influence based on its position in the curriculum:

$$\gamma_i^r = \begin{cases} 1, & \text{if } i \text{ has been in } S_r \text{ for more than } \rho \text{ rounds} \\ \exp\left(\frac{r_i - r + \rho}{\rho}\right), & \text{if } r - r_i < \rho \end{cases} \quad (14)$$

where r_i denotes the round at which client i was first included in the active client set, and ρ is a hyperparameter controlling the adaptation period. This formulation implements a warm-up phase for newly added clients, gradually increasing their influence on the global model as they integrate into the curriculum. To further enhance optimization stability during curriculum transitions, we implement a regularization term that prevents catastrophic forgetting when new clients are added:

$$F_i^r(\mathbf{w}) = \frac{1}{|D_i|} \sum_{(x, y) \in D_i} \mathcal{L}(\mathbf{w}, x, y) + \mu_r \|\mathbf{w} - \mathbf{w}^{r-1}\|^2 \quad (15)$$

where μ_r is an adaptive regularization parameter that increases temporarily when new clients are added to the curriculum, then gradually decreases as the model adapts to the expanded client set:

$$\mu_r = \begin{cases} \mu_0 \cdot \exp\left(-\frac{r-r^*}{\kappa}\right), & \text{if } |S_r| > |S_{r-1}| \text{ and } r - r^* < \kappa, \\ \mu_{\text{base}}, & \text{otherwise.} \end{cases} \quad (16)$$

where r^* is the most recent round at which a new client was added, κ controls the decay rate, and μ_0, μ_{base} are hyperparameters defining the initial and baseline regularization strengths.

4.3. Comparison with standard FL approaches (FedAvg, FedProx, FedCurv)

Our Sequential Federated Learning (SFL) approach with curriculum learning differs fundamentally from standard federated learning algorithms in both client selection strategy and optimization dynamics. Here, we provide a formal comparison with several prominent federated learning approaches: **FedAvg**: The canonical FedAvg algorithm updates the global model using weighted averaging of client models:

$$\mathbf{w}^{r+1}_{\text{FedAvg}} = \sum_{i=1}^N \frac{|D_i|}{\sum_{j=1}^N |D_j|} \mathbf{w}_i^r \quad (17)$$

FedAvg assumes all clients participate in each round or randomly selects a subset of clients. In contrast, our SFL approach uses a deterministic, performance-based client selection:

$$\mathbf{w}^{r+1}_{\text{SFL}} = \sum_{i \in S_r} \beta_i^r \mathbf{w}_i^r \quad (18)$$

where S_r is progressively expanded based on our curriculum function Φ_r and β_i^r incorporates class diversity and curriculum stage.

FedProx: FedProx addresses client drift through proximal term regularization:

$$\mathbf{w}^r = \arg \min_{\mathbf{w}} \left[F_i(\mathbf{w}) + \frac{\mu}{2} \|\mathbf{w} - \mathbf{w}^r\|^2 \right] \quad (19)$$

While FedProx uses a fixed regularization coefficient μ , our approach implements an adaptive regularization μ_r that responds to curriculum transitions. Additionally, FedProx lacks the progressive client integration mechanism central to our approach.

FedCurv: The FedCurv approach incorporates importance weights based on Fisher information matrices to prevent catastrophic forgetting:

$$\mathbf{w}_{\text{FedCurv}}^{r+1} = \mathbf{w}^r - \eta \sum_{i=1}^N \frac{|D_i|}{\sum_{j=1}^N |D_j|} \nabla F_i(\mathbf{w}^r) - \lambda \sum_k F_k \odot (\mathbf{w}^r - \mathbf{w}^*) \quad (20)$$

where F_k represents the diagonal of the Fisher information matrix for parameter k , and $\mathbf{w}^*$ represents previous model parameters. While FedCurv focuses on preserving knowledge from past tasks, our SFL approach prioritizes progressive knowledge acquisition through curriculum-based client integration.

FedCL: The Federated Curriculum Learning approach proposed in prior work implements a data-level curriculum within each client, prioritizing "easier" examples during early training:

$$\mathcal{L}_{\text{FedCL}}(\mathbf{w}, D_i) = \sum_{(x, y) \in D_i} v_{(x, y)}^r \cdot \mathcal{L}(\mathbf{w}, x, y) \quad (21)$$

where $v_{(x, y)}^r$ represents the sample weight at round r . In contrast, our approach implements a client-level curriculum that progressively expands the participating client set based on model performance, focusing on the heterogeneity of data distributions across clients rather than individual samples.

5. Experimental Setup

5.1. FL system architecture (server-client interaction)

Our sequential federated learning (SFL) system implements a hub-and-spoke architecture with a central server orchestrating the training process across multiple clients. The system follows a synchronous communication protocol where parameter updates occur in discrete federated rounds. Formally, we define the server-client interaction as a tuple $(S, C, D, \mathcal{P})$, where S represents the server, $C = \{1, 2, \dots, N\}$ denotes the set of N clients, $D = \{D_1, D_2, \dots, D_N\}$ represents the local datasets, and $\mathcal{P}$ defines the communication protocol.

At each federated round r , the server-client interaction follows a four-phase protocol:

1. **Client Selection**: The server determines the active client set $S_r \subseteq C$ based on the curriculum function Φ_r , as defined in Section 5.
2. **Model Distribution**: The server broadcasts the current global model parameters $\mathbf{w}^r$ to all active clients $i \in S_r$.

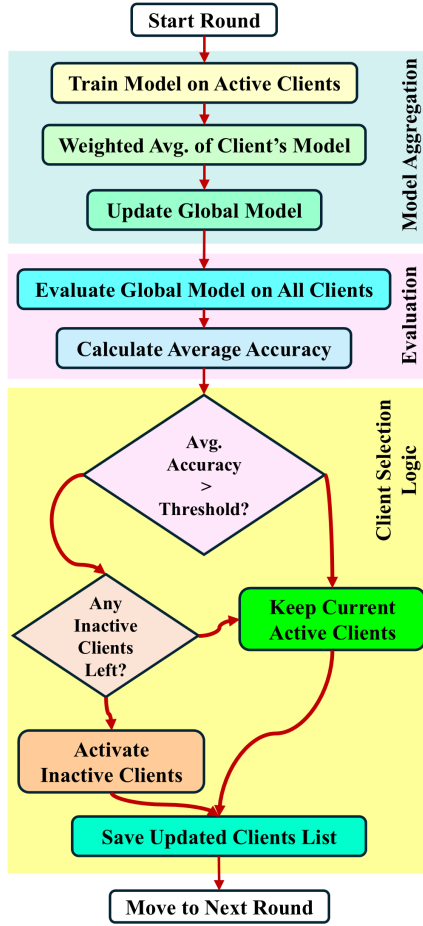


Figure 5: Flow Chart for the Description of Learning.

3. **Local Training:** Each active client i performs E epochs of local training, minimizing:

$$F_i^r(\mathbf{w}) = \frac{1}{|\mathcal{D}_i|} \sum_{(x,y) \in \mathcal{D}_i} \mathcal{L}(\mathbf{w}, x, y) + \mu_r \|\mathbf{w} - \mathbf{w}^r\|^2 \quad (22)$$

using mini-batch stochastic gradient descent with learning rate η_{local} , resulting in updated local parameters $\mathbf{w}_i^r$.

4. **Aggregation:** Clients transmit their model parameters $\mathbf{w}_i^r$ to the server, which aggregates them using the curriculum-aware weighting scheme:

$$\mathbf{w}^{r+1} = \sum_{i \in S_r} \beta_i^r \mathbf{w}_i^r \quad (23)$$

where β_i^r represents the curriculum-adaptive weight for client i at round r .

The communication complexity per round is $\mathcal{O}(|S_r| \cdot |\mathbf{w}|)$, where $|\mathbf{w}|$ denotes the number of model parameters. Notably, our SFL approach reduces overall communication overhead compared to traditional FL by limiting $|S_r|$ in early rounds and incrementally increasing client participation based on performance thresholds.

5.2. CHARM (Channel-Hierarchical Action Recognition Model)

We propose a novel deep learning architecture for video action recognition that incorporates 3D convolutions with residual connections and attention mechanisms, specifically designed for federated learning environments. Our model addresses the challenges of sequential knowledge transfer across heterogeneous clients.

The architecture consists of multiple components that process spatio-temporal information from video sequences. The input to our model is a tensor $\mathbf{X} \in \mathbb{R}^{C \times T \times H \times W}$, where C represents channels (3 for RGB videos), T denotes the temporal dimension (sequence length), and $H \times W$ represents the spatial dimensions.

5.2.1. 3D Convolutional Stem

The initial feature extraction is performed by a 3D convolutional stem:

$$\mathbf{F}_0 = \text{Conv3D}(\mathbf{X}; \theta_{\text{conv1}}) \quad (24)$$

$$\mathbf{F}_0 = \text{BatchNorm3D}(\mathbf{F}_0; \theta_{\text{bn1}}) \quad (25)$$

$$\mathbf{F}_0 = \text{ReLU}(\mathbf{F}_0) \quad (26)$$

$$\mathbf{F}_0 = \text{MaxPool3D}(\mathbf{F}_0) \quad (27)$$

where $\mathbf{F}_0 \in \mathbb{R}^{64 \times T \times \frac{H}{2} \times \frac{W}{2}}$ is the initial feature map.

5.2.2. Residual Blocks with Attention

We implement 3D residual blocks with dual attention mechanisms (channel and temporal attention). Each residual block performs:

$$\begin{aligned} \mathbf{F}_{\text{out}} &= \text{ResidualBlock3D}(\mathbf{F}_{\text{in}}) \\ &= \mathbf{F}_{\text{in}} + \text{Attention}(\text{Conv3D}_2(\text{ReLU}(\text{BatchNorm3D}(\text{Conv3D}_1(\mathbf{F}_{\text{in}})))) \end{aligned} \quad (28)$$

The dual attention mechanism operates as follows:

Channel Attention:

$$\mathbf{Y}_{\text{channel}} = \text{SpatialAvgPool}(\mathbf{F}) \in \mathbb{R}^{B \times C \times T \times 1 \times 1}$$

$$\mathbf{Y}_{\text{channel}} = \frac{1}{T} \sum_{t=1}^T \mathbf{Y}_{\text{channel}}(:, :, t, :, :) \quad (29)$$

$$\mathbf{Y}_{\text{channel}} = \sigma(W_2 \text{ReLU}(W_1 \mathbf{Y}_{\text{channel}}))$$

Temporal Attention:

$$\mathbf{Y}_{\text{temporal}} = \text{TemporalAvgPool}(\mathbf{F}) \in \mathbb{R}^{B \times C \times 1 \times H \times W}$$

$$\mathbf{Y}_{\text{temporal}} = \sigma(\text{Conv3D}_2(\text{ReLU}(\text{Conv3D}_1(\mathbf{Y}_{\text{temporal}})))) \quad (30)$$

The final attention-modulated feature is:

$$\mathbf{F}_{\text{att}} = \mathbf{F} \odot \mathbf{Y}_{\text{channel}} \odot \mathbf{Y}_{\text{temporal}} \quad (31)$$

where $\odot$ represents element-wise multiplication with broadcasting.

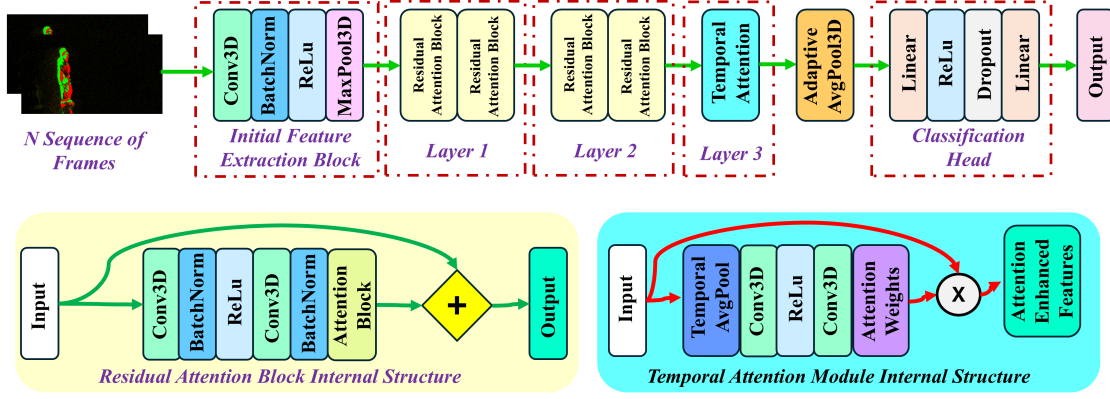


Figure 6: Detailed Architecture of the Proposed CHARM Model used in Sequential Federated Learning.

5.2.3. Temporal Attention after Feature Extraction

After the residual blocks, we apply another temporal attention mechanism:

$$\begin{aligned} \mathbf{A}_{\text{temporal}} &= \sigma(\text{Conv3D}(\mathbf{F}_{\text{last}}; \theta_{\text{temp}})) \\ \mathbf{F}_{\text{att}} &= \mathbf{F}_{\text{last}} \odot \mathbf{A}_{\text{temporal}} \end{aligned} \quad (32)$$

5.2.4. Classification Head

The global average pooling and classification head perform:

$$\mathbf{z} = \text{AvgPool3D}(\mathbf{F}_{\text{att}}) \quad (33)$$

$$\mathbf{z} = \text{Flatten}(\mathbf{z}) \quad (34)$$

$$\mathbf{h} = \text{ReLU}(W_{\text{fc1}} \mathbf{z}) \quad (35)$$

$$\mathbf{h} = \text{Dropout}(\mathbf{h}, p = 0.5) \quad (36)$$

$$\mathbf{y} = W_{\text{fc2}} \mathbf{h} \quad (37)$$

where $\mathbf{y} \in \mathbb{R}^{C_{\text{classes}}}$ represents the logits for each action class.

5.3. Evaluation metrics

We employ a comprehensive suite of evaluation metrics to assess both the global model performance and the effectiveness of our curriculum learning approach. For model performance evaluation, we utilize:

1. **Accuracy** is the most straightforward and intuitive metric, measuring the proportion of correct predictions among all predictions made. In mathematical terms, accuracy is defined as:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \quad (38)$$

To evaluate federated learning and curriculum effectiveness, we introduce specialized metrics:

1. **Client Convergence Rate (CCR)**: Measures how quickly each client reaches a target performance threshold τ_{target} :

$$\text{CCR}_i = \min\{r : A_i(\mathbf{w}^r, \mathcal{D}_i^{\text{test}}) \geq \tau_{\text{target}}\} \quad (39)$$

where $A_i(\mathbf{w}^r, \mathcal{D}_i^{\text{test}})$ represents the test accuracy for client i using model $\mathbf{w}^r$.

Algorithm 1 Dual-Attention Block (Channel + Temporal Attention)

```

1: procedure ATTENTIONBLOCK( $X \in \mathbb{R}^{B \times C \times T \times H \times W}$ )
2:   Input: Feature tensor  $X$  with batch size  $B$ , channels  $C$ , temporal  $T$ , height  $H$ , width  $W$ 
3:    $r \leftarrow 8$  ▷ Reduction ratio
4:   // Channel Attention Branch
5:    $X_{\text{spatial}} \leftarrow \text{AdaptiveAvgPool3D}(X, (T, 1, 1))$  ▷ Pool spatial dimensions
6:    $X_{\text{channel}} \leftarrow \text{Mean}(X_{\text{spatial}}, \text{dim} = 2)$  ▷ Average over temporal dimension
7:    $X_{\text{channel}} \leftarrow \text{Linear}(X_{\text{channel}}, C \rightarrow C/r)$ 
8:    $X_{\text{channel}} \leftarrow \text{ReLU}(X_{\text{channel}})$ 
9:    $X_{\text{channel}} \leftarrow \text{Linear}(X_{\text{channel}}, C/r \rightarrow C)$ 
10:   $A_{\text{channel}} \leftarrow \text{Sigmoid}(X_{\text{channel}})$  ▷ Channel attention weights
11:   $A_{\text{channel}} \leftarrow \text{Reshape}(A_{\text{channel}}, (B, C, 1, 1, 1))$ 
12:  // Temporal Attention Branch
13:   $X_{\text{temporal}} \leftarrow \text{AdaptiveAvgPool3D}(X, (1, H, W))$  ▷ Pool temporal dimension
14:   $X_{\text{temporal}} \leftarrow \text{Conv3D}(X_{\text{temporal}}, C \rightarrow C/r, \text{kernel} = 1)$ 
15:   $X_{\text{temporal}} \leftarrow \text{ReLU}(X_{\text{temporal}})$ 
16:   $X_{\text{temporal}} \leftarrow \text{Conv3D}(X_{\text{temporal}}, C/r \rightarrow C, \text{kernel} = 1)$ 
17:   $A_{\text{temporal}} \leftarrow \text{Sigmoid}(X_{\text{temporal}})$  ▷ Temporal attention weights
18:  // Apply Dual Attention
19:   $X_{\text{out}} \leftarrow X \odot A_{\text{channel}} \odot A_{\text{temporal}}$  ▷ Element-wise multiplication
20:  return  $X_{\text{out}}$ 
21: end procedure
    
```

2. **Knowledge Transfer Efficiency (KTE)**: Quantifies how effectively knowledge transfers from existing clients to newly added clients:

$$\text{KTE}_i = \frac{A_i(\mathbf{w}^r, \mathcal{D}_i^{\text{test}}) - A_i(\mathbf{w}^0, \mathcal{D}_i^{\text{test}})}{A_i(\mathbf{w}^{\text{local}}, \mathcal{D}_i^{\text{test}}) - A_i(\mathbf{w}^0, \mathcal{D}_i^{\text{test}})} \quad (40)$$

Algorithm 2 3D Residual Block with Integrated Attention

```

1: procedure RESIDUALBLOCK3D( $X \in \mathbb{R}^{B \times C_{in} \times T \times H \times W}, C_{out}, s$ )  $\in$ 
2:   Input: Feature tensor  $X$ , output channels  $C_{out}$ , stride  $s$ 
3:   // Main Branch
4:    $F \leftarrow \text{Conv3D}(X, C_{in} \rightarrow C_{out}, \text{kernel} = 3, \text{stride} = s, \text{padding} = 1)$ 
5:    $F \leftarrow \text{BatchNorm3D}(F)$ 
6:    $F \leftarrow \text{ReLU}(F)$ 
7:    $F \leftarrow \text{Conv3D}(F, C_{out} \rightarrow C_{out}, \text{kernel} = 3, \text{stride} = 1, \text{padding} = 1)$ 
8:    $F \leftarrow \text{BatchNorm3D}(F)$ 
9:   // Apply Attention to Features
10:   $F \leftarrow \text{AttentionBlock}(F)$   $\triangleright$  Dual attention mechanism
11:  // Shortcut Connection
12:  if  $s \neq 1$  or  $C_{in} \neq C_{out}$  then
13:     $X_{shortcut} \leftarrow \text{Conv3D}(X, C_{in} \rightarrow C_{out}, \text{kernel} = 1, \text{stride} = s)$ 
14:     $X_{shortcut} \leftarrow \text{BatchNorm3D}(X_{shortcut})$ 
15:  else
16:     $X_{shortcut} \leftarrow X$ 
17:  end if
18:  // Residual Addition
19:   $Y \leftarrow F + X_{shortcut}$ 
20:   $Y \leftarrow \text{ReLU}(Y)$ 
21:  return  $Y$ 
22: end procedure

```

where r_i denotes the round at which client i is first added to the active set, $\mathbf{w}^0$ represents the initial model parameters, and $\mathbf{w}^{\text{local}}$ represents client i 's locally trained model (without federation).

3. **Catastrophic Forgetting Metric (CFM):** Measures performance degradation on previously learned classes when new clients are added:

$$\text{CFM}(r, c) = \frac{A_c(\mathbf{w}^{r-1}, D_c^{\text{test}}) - A_c(\mathbf{w}^r, D_c^{\text{test}})}{A_c(\mathbf{w}^{r-1}, D_c^{\text{test}})} \quad (41)$$

where $A_c(\mathbf{w}, D)$ denotes the accuracy for class c using model $\mathbf{w}$ on dataset D .

4. **Communication Efficiency (CE):** Quantifies the reduction in communication overhead compared to standard FL:

$$\text{CE} = \frac{\sum_{r=1}^R N \cdot |\mathbf{w}|}{\sum_{r=1}^R |S_r| \cdot |\mathbf{w}|} = \frac{N \cdot R}{\sum_{r=1}^R |S_r|} \quad (42)$$

where N is the total number of clients and $|\mathbf{w}|$ is the model size.

These metrics collectively provide a comprehensive evaluation of both the model's action recognition performance and the efficacy of our sequential federated learning approach with curriculum learning.

Algorithm 3 Enhanced Action Recognition Model Forward Pass

```

1: procedure ENHANCEDACTIONRECOGNITIONMODEL( $X \in \mathbb{R}^{B \times 3 \times T \times H \times W}$ )
2:   Input: Video sequence  $X$  with 3 RGB channels,  $T$  frames
3:   Output: Action class predictions  $\in \mathbb{R}^{B \times 11}$ 
4:   // Initial 3D Convolution Layer
5:    $X \leftarrow \text{Conv3D}(X, 3 \rightarrow 64, \text{kernel} = 7, \text{stride} = (1, 2, 2), \text{padding} = 3)$ 
6:    $X \leftarrow \text{BatchNorm3D}(X)$ 
7:    $X \leftarrow \text{ReLU}(X)$ 
8:    $X \leftarrow \text{MaxPool3D}(X, \text{kernel} = (1, 3, 3), \text{stride} = (1, 2, 2))$ 
9:   // Layer 1: Residual Blocks with Attention
10:  for  $i = 1$  to 2 do
11:     $X \leftarrow \text{ResidualBlock3D}(X, C_{out} = 64, s = 1)$ 
12:  end for
13:  // Layer 2: Residual Blocks with Attention and Downsampling
14:  for  $i = 1$  to 2 do
15:    if  $i = 1$  then
16:       $X \leftarrow \text{ResidualBlock3D}(X, C_{out} = 128, s = (1, 2, 2))$   $\triangleright$  Spatial downsampling
17:    else
18:       $X \leftarrow \text{ResidualBlock3D}(X, C_{out} = 128, s = 1)$ 
19:    end if
20:  end for
21:  // Global Temporal Attention
22:   $A_{global} \leftarrow \text{Conv3D}(X, 128 \rightarrow 1, \text{kernel} = 1)$ 
23:   $A_{global} \leftarrow \text{Sigmoid}(A_{global})$   $\triangleright$  Global temporal attention map
24:   $X \leftarrow X \odot A_{global}$   $\triangleright$  Apply global temporal weighting
25:  // Global Average Pooling
26:   $X \leftarrow \text{AdaptiveAvgPool3D}(X, (1, 1, 1))$   $\triangleright$  Output:  $B \times 128 \times 1 \times 1 \times 1$ 
27:   $X \leftarrow \text{Flatten}(X)$   $\triangleright$  Output:  $B \times 128$ 
28:  // Classification Head
29:   $X \leftarrow \text{Linear}(X, 128 \rightarrow 128)$ 
30:   $X \leftarrow \text{ReLU}(X)$ 
31:   $X \leftarrow \text{Dropout}(X, p = 0.5)$ 
32:   $X \leftarrow \text{Linear}(X, 128 \rightarrow 11)$   $\triangleright$  11 action classes
33:  return  $X$ 
34: end procedure

```

5.4. Hyperparameters and training configurations

Our experimental setup employs carefully tuned hyperparameters optimized for the action recognition task in a federated setting. The configuration parameters can be categorized into model-specific, federated learning, and curriculum learning hyperparameters:

The training process is initialized with random weights drawn from a Kaiming normal distribution for convolutional

Table 3
Hyperparameters of the Proposed Enhanced Action Recognition Model

Component	Parameter	Description
Input Processing	Input dimensions	$3 \times 8 \times 64 \times 64$ (channels, time steps, height, width)
	Sequence length	8 frames per action sequence
	Image size	64×64 pixels
Network Architecture	Conv1 filters	64 filters with $7 \times 7 \times 7$ kernel, stride (1, 2, 2)
	Layer1	2 residual blocks with 64 channels
	Layer2	2 residual blocks with 128 channels, stride (1, 2, 2)
	Attention mechanisms	Channel attention and temporal attention
	Attention reduction ratio	8 (for dimensionality reduction in attention blocks)
	FC layers	128 neurons with ReLU, followed by output layer
Training Parameters	Output layer	11 neurons (one per action class)
	Optimizer	AdamW
	Learning rate	1×10^{-4}
	Batch size	16
	Weight decay	1×10^{-5}
	Dropout rate	0.5 (applied in fully connected layers)
Federated Learning	Weight initialization	Kaiming normal (conv layers), uniform (linear layers)
	Client epochs	10 local training epochs per round
	Federated rounds	15 global aggregation rounds
	Client weighting	Based on class diversity ($\alpha_i = \frac{ C_i }{\sum_{j \in S_r} C_j }$)
	Performance threshold	0.65 (for adding new clients)
	Regularization	Proximal term with adaptive coefficient

Table 4
Hyperparameters of the Sequential Federated Learning Framework

Parameter	Value	Description
Number of clients (N)	8 for E-FedAct and 15 for PAF	Total clients in the federation
Total federated rounds (R)	15	Maximum global rounds
Client epochs (E)	10	Local training epochs per round
Proximal term coefficient (μ_{base})	0.01	Baseline regularization strength
Initial proximal term (μ_0)	0.1	Initial regularization for new clients
Proximal term decay rate (κ)	3	Controls regularization decay

layers and a uniform distribution for linear layers. We employ a train-test split of 80%-20% for each client's local dataset, ensuring consistent evaluation across all experiments. To account for stochastic variations, we conduct each experiment with three different random seeds and report mean performance metrics with standard deviations.

The sequential client integration follows our curriculum design, starting with Client 1 (containing all action classes) and progressively adding clients based on performance thresholds. The initial performance threshold τ is set to 65%, requiring the global model to achieve this accuracy

averaged across all active clients before incorporating the next client in the sequence. This threshold provides a balance between rapid curriculum progression and adequate knowledge consolidation.

All experiments are conducted in a controlled environment to ensure fair comparison, with consistent hardware configurations across all clients and identical data preprocessing pipelines. The implementation uses PyTorch 1.10.0 with CUDA 11.3 for GPU acceleration, and all timing measurements exclude communication overhead to focus on computational efficiency.

Table 5
Hyperparameters of the Circullum Learning in SFL

Parameter	Value	Description
Performance threshold (τ)	0.65	Accuracy threshold for adding clients
Min classes per client (m)	3	Minimum action classes per client
Adaptation period (ρ)	3	Rounds for gradual client integration
Class weight factor (λ_1)	0.5	Weight for class uniqueness in difficulty score
Distribution entropy factor (λ_2)	0.3	Weight for class distribution entropy
Domain gap factor (λ_3)	0.2	Weight for domain gap measurement

6. Results and Discussion

6.1. Ablation Study

The ablation study, presented in Table 6 and Table 7, systematically investigates the contribution of each architectural component to the performance of our enhanced action recognition model within the proposed curriculum learning-based federated learning framework across two benchmark datasets. The model consists of three key components: Layer 1, Layer 2 and Layer 3.

On the E-FedAct dataset, the baseline configuration S1, which incorporates all three layers, achieves the highest performance at 94.87% accuracy, demonstrating the effectiveness of our complete architecture. When Layer 1 is removed (S2), there is a modest performance drop to 93.31% (1.56 percentage point decrease), suggesting that while the initial residual block contributes to model performance, higher-level features captured by Layer 2 and the temporal dynamics identified by Layer 3 carry more discriminative power. Most notably, eliminating the Global Temporal Attention mechanism (S3) results in a substantial decrease to 85.19% accuracy (9.68 percentage point drop), underscoring the critical importance of temporal dynamics modeling in action recognition tasks. The most minimal configuration (S4), retaining only Layer 2, performs marginally worse at 84.61%, indicating that the middle-level feature extraction remains a fundamental component even in reduced architectures.

The PAF dataset demonstrates remarkably consistent trends with even more pronounced performance differences. The complete model (S1) achieves 95.59% accuracy, representing the highest performance across both datasets. Removing Layer 1 (S2) results in a 1.60 percentage point decrease to 93.99%, closely mirroring the E-FedAct results and confirming the modest but consistent contribution of the initial layer. The elimination of Layer 3 (S3) yields an even more dramatic performance degradation to 87.17%, representing an 8.42 percentage point drop, which reinforces the critical role of temporal attention mechanisms. The minimal configuration (S4) achieves 83.91% accuracy, showing a 3.26 percentage point further decrease from S3, highlighting the foundational importance of Layer 2's feature extraction capabilities.

Cross-dataset analysis reveals that while both datasets exhibit similar architectural component importance rankings (Layer 3 > Layer 1 > baseline Layer 2 dependency), the PAF dataset shows slightly higher baseline performance but maintains consistent relative performance gaps. These findings provide compelling evidence that our proposed Global Temporal Attention mechanism significantly enhances the model's ability to focus on discriminative frames within video sequences across diverse action recognition scenarios, which is particularly valuable in the heterogeneous data environments characteristic of federated learning for human action recognition.

The federated learning metrics, shown in Fig. 7, and Fig. 8, reveal compelling insights about our model architecture's learning dynamics across both datasets. On the

Table 6

Ablation Study of the Proposed Model using Curriculum Learning based Federated Learning Framework with 65% Threshold and 10 Epochs for each client on E-FedAct dataset

Scenario	Layer 3	Layer 2	Layer 1	Accuracy (%)
S1	✓	✓	✓	94.87
S2	✓	✓	×	93.31
S3	×	✓	✓	85.19
S4	×	✓	×	84.61

Table 7

Ablation Study of the Proposed Model using Curriculum Learning based Federated Learning Framework with 65% Threshold and 20 Epochs for each client on PAF dataset

Scenario	Layer 3	Layer 2	Layer 1	Accuracy (%)
S1	✓	✓	✓	95.59
S2	✓	✓	×	93.99
S3	×	✓	✓	87.17
S4	×	✓	×	83.91

E-FedAct dataset, Client Convergence Rate (CCR) values show a clear advantage for the complete model (S1) with the fastest convergence at 0.9275, followed by progressively slower rates for reduced architectures: S2 (0.9350), S3 (0.9500), and S4 (0.9620). The PAF dataset demonstrates even more pronounced convergence benefits, with S1 achieving the fastest rate at 0.9010, followed by S2 (0.9120), S3 (0.9330), and S4 (0.9545), indicating that the complete architecture provides superior convergence efficiency across different data characteristics. However, the most significant finding comes from Knowledge Transfer Efficiency (KTE) data, where configurations including the Global Temporal Attention mechanism consistently demonstrate substantially higher transfer efficiency across both datasets. On E-FedAct, S1 and S2 achieve 0.6595 and 0.6172 respectively, compared to S3 and S4 at 0.2500 and 0.2400, representing approximately 60% reduction in transfer capability. The PAF dataset shows an even more dramatic pattern, with S1 and S2 reaching 0.7260 and 0.6480 respectively, while S3 and S4 drop significantly to 0.3510 and 0.3200, indicating a 52-56% reduction in knowledge transfer efficiency when Layer 3 is removed. This cross-dataset consistency in the critical role of temporal attention emphasizes that the Global Temporal Attention mechanism fundamentally enhances the federated learning process by facilitating the transfer of generalizable temporal patterns across clients, making it particularly valuable for action recognition tasks in distributed settings regardless of the underlying data characteristics.

The client contribution weights, shown in Fig. 10, in proposed framework demonstrate a strategic progression of knowledge accumulation across training rounds. Beginning with only two clients (weights 0.55 and 0.45) in round 3, we observe a gradual redistribution of influence as additional clients join the federation, ultimately culminating in a more balanced eight-client collaboration by round 9 (weights ranging from 0.055 to 0.200). Notably, Client 1,

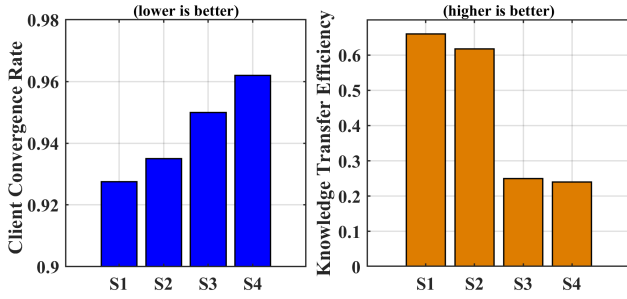


Figure 7: Average Client Convergence Rate and Average Knowledge Transfer Efficiency for different models on E-FedAct Dataset.

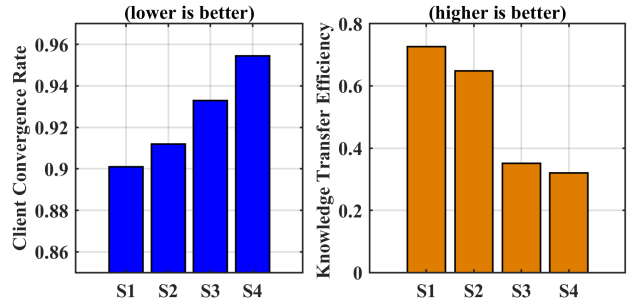


Figure 8: Average Client Convergence Rate and Average Knowledge Transfer Efficiency for different models on PAF Dataset.

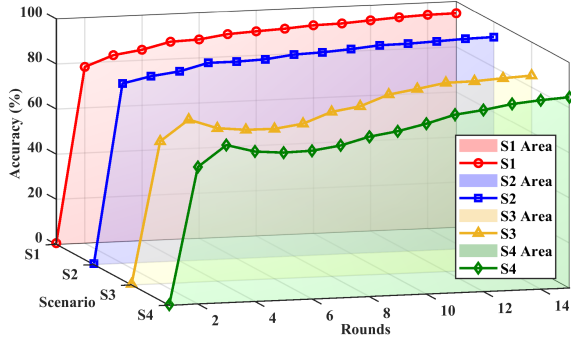


Figure 9: Round-wise Test Accuracy of the different Models.

which contains all action classes, maintains the highest contribution weight throughout the entire training process, declining from 0.55 to 0.200 as new clients are incorporated. This weighting scheme, based proportionally on each client's class representation, ensures that clients with richer and more diverse datasets exert greater influence on the global model while still allowing specialized clients to contribute their domain expertise. The progressive integration of clients with varying class distributions creates a curriculum effect where the model first learns comprehensive representations from data-rich clients before refining its understanding with more specialized data streams. This approach mitigates catastrophic forgetting while maximizing knowledge transfer across the heterogeneous client ecosystem, thereby

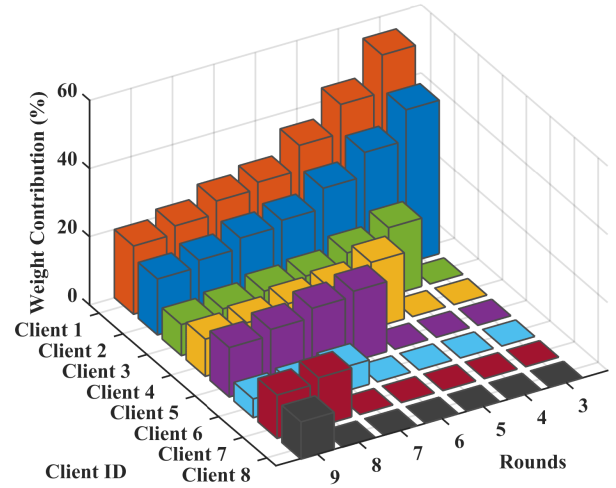


Figure 10: Training Weights distribution of all clients over the rounds for the E-FedAct dataset.

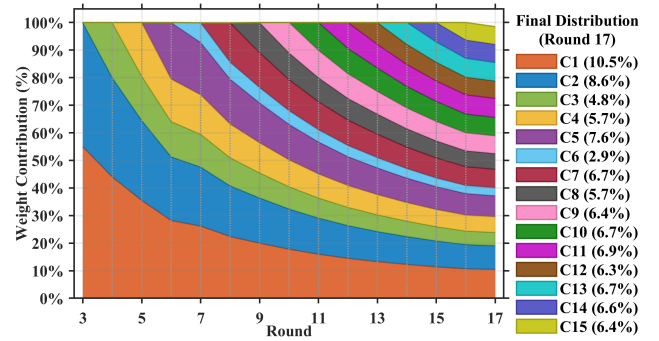


Figure 11: Training Weights distribution of all clients over the rounds for the PAF dataset.

enhancing the model's ability to generalize across the full spectrum of human action recognition tasks.

6.2. Dynamic Training Epochs Analysis

An analysis of the Curriculum Federated Learning (Cur_Fed) performance across different epoch configurations reveals significant patterns in convergence behaviour and ultimate model accuracy. The results demonstrate a clear correlation between the number of local training epochs and both the learning speed and final performance achieved. The 5-epoch configuration shows the slowest start with just 0.13% accuracy in Round 1, but exhibits a dramatic improvement to 47.96% by Round 2. This configuration reaches a relatively stable plateau around Round 8, ultimately achieving 90.39% accuracy by Round 15. The early learning trajectory shows notable volatility, with a notable dip in Round 5 (78.58% down from 84.16% in Round 4), suggesting potential instability when using fewer epochs per client. The 10-epoch configuration demonstrates improved initial learning with 0.46% accuracy in Round 1 and 58.92% in Round 2, outperforming the 5-epoch model by approximately 11 percentage points in early training. This configuration shows more consistent improvement without

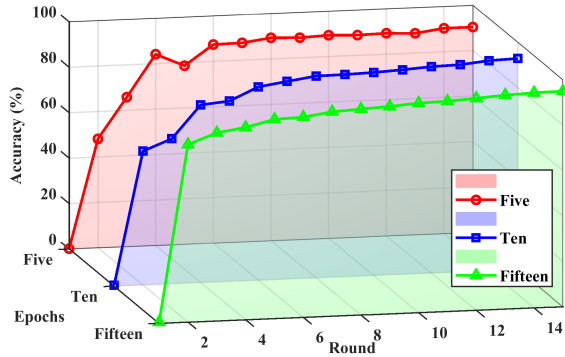


Figure 12: Comparison of Test accuracies for different training epochs.

significant performance drops, reaching 92.94% by the final round - a 2.55 percentage point improvement over the 5-epoch model. Most notably, the 15-epoch configuration exhibits substantially superior performance across all stages of training. Starting with 0.55% accuracy in Round 1, it makes a remarkable leap to 78.15% in Round 2 (approximately 30 percentage points higher than the 5-epoch configuration). This configuration maintains accelerated learning throughout all rounds, achieving the highest final accuracy of 94.87%. These results highlight the critical importance of sufficient local training in Curriculum Federated Learning. The 15-epoch configuration allows clients to better optimize local objectives before model aggregation, leading to more informative updates to the global model. The performance gains are particularly pronounced in early rounds, suggesting that more extensive local training enables better feature extraction and knowledge transfer when new clients are introduced into the federation. This observation aligns with the theoretical foundation of curriculum learning, where establishing a robust base of knowledge before tackling more complex patterns significantly enhances overall model convergence and generalization capabilities.

6.3. Federated Learning Techniques

The comparative analysis of federated learning approaches across both E-FedAct and PAF datasets reveals that Curriculum Federated Learning (Cur-Fed) demonstrates consistently superior performance, achieving the highest average accuracy of 94.87% on E-FedAct and 95.59% on PAF while maintaining remarkable consistency across all clients. This methodological advantage stems from Cur-Fed's progressive learning strategy, which systematically introduces clients based on performance thresholds, allowing the global model to establish a strong foundation before tackling more challenging or diverse data distributions. The enhanced performance on PAF dataset (0.72 percentage point improvement) indicates that Cur-Fed's structured approach becomes even more effective with larger client populations, as evidenced by its ability to handle 15 clients compared to 8 on E-FedAct while maintaining superior individual client performance. In contrast, FedAvg shows respectable overall performance on both datasets (92.99% on E-FedAct, 92.41%

on PAF) but exhibits notable weaknesses with specific clients—particularly Client 8 on both datasets (85.31% and 86.79% respectively)—suggesting consistent limitations in addressing significant statistical heterogeneity across clients without curriculum-based scheduling.

FedProx demonstrates the most inconsistent behavior across both datasets, despite incorporating a proximal term to constrain local updates. On E-FedAct, it shows high variability with the best individual client result (97.18% on Client 5) while simultaneously exhibiting substantial degradation on Clients 3 and 8 (85.15% and 86.48% respectively). The PAF dataset reveals even more pronounced inconsistencies, with FedProx achieving excellent performance on select clients (96.85% on Client 11, 96.42% on Client 5) but showing severe degradation on multiple clients including Client 3 (87.93%), Client 4 (89.54%), and Client 8 (87.21%). This pattern indicates that while proximal regularization helps certain clients with specific data characteristics, it may overly constrain others, leading to a 0.06 percentage point decrease in average performance from E-FedAct to PAF (92.42% to 92.36%). FedCurv consistently demonstrates the most stable but lowest performance across both datasets (92.09% on E-FedAct, 91.79% on PAF), with its Fisher Information Matrix-based regularization appearing too conservative in allowing client-specific optimization, particularly as the number of clients increases.

The cross-dataset analysis highlights how Curriculum Federated Learning effectively balances knowledge accumulation and transfer through its structured participation schedule, enabling stronger performance on all clients regardless of their data distribution peculiarities or the scale of federated deployment. This approach proves particularly valuable in non-IID settings where traditional federated methods struggle to reconcile global model coherence with client-specific optimization needs. The enhanced performance of Cur-Fed can be attributed to its ability to build knowledge incrementally, allowing the model to first learn generalizable features from clients with broader class coverage before adapting to more specialized data distributions, thereby mitigating catastrophic forgetting while maximizing knowledge transfer. The consistent superiority across both 8-client and 15-client scenarios demonstrates the scalability and robustness of the curriculum-based approach, making it especially suitable for real-world federated learning deployments where client heterogeneity and varying participation patterns are common challenges.

Our analysis of communication costs, shown in Fig. 14, reveals a significant advantage of our Curriculum Federated Learning (Cur Fed) approach compared to traditional federated learning methods. While FedAvg, FedProx, and FedCurv all exhibit identical communication costs of 2056.31 MB, our proposed Cur Fed technique substantially reduces this overhead to 1494.04 MB—representing a 27.3% decrease in communication requirements. This improvement stems from our sequential client integration strategy, which strategically adds clients only after the model reaches a performance threshold, thereby limiting unnecessary data

Table 8

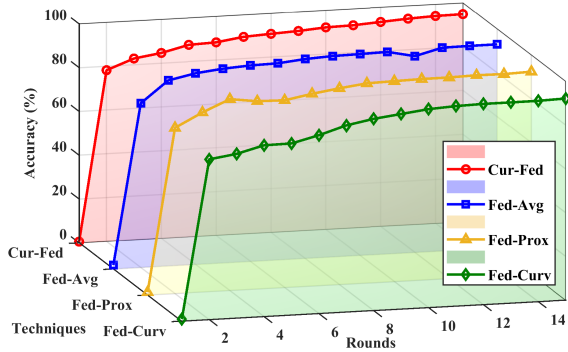
Client-wise test accuracy comparison of different federated learning techniques on E-FedAct Dataset

Client	Cur Fed	Fed Avg	Fed Prox	Fed Curv
Client 1	96.44%	95.22%	95.61%	94.08%
Client 2	92.39%	94.67%	92.87%	92.17%
Client 3	94.19%	93.90%	85.15%	93.48%
Client 4	96.84%	95.18%	90.98%	92.22%
Client 5	95.03%	93.96%	97.18%	93.07%
Client 6	94.43%	93.40%	96.03%	92.03%
Client 7	95.52%	92.20%	95.29%	91.29%
Client 8	94.09%	85.31%	86.48%	88.36%
Avg.	94.87%	92.99%	92.42%	92.09%

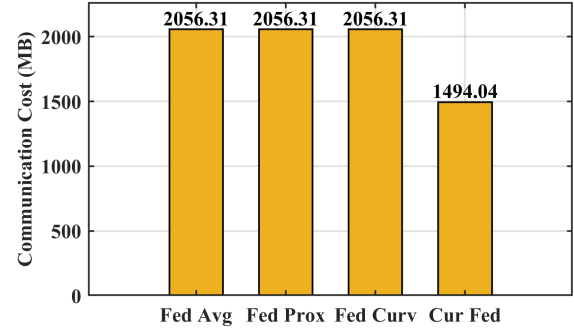
Table 9

Client-wise test accuracy comparison of different federated learning techniques on PAF dataset

Client	Cur Fed	Fed Avg	Fed Prox	Fed Curv
Client 1	96.85%	93.85%	94.29%	93.41%
Client 2	92.94%	93.24%	91.56%	90.88%
Client 3	94.79%	92.48%	87.93%	92.75%
Client 4	98.27%	94.76%	89.54%	91.63%
Client 5	95.70%	92.73%	96.42%	92.84%
Client 6	95.03%	92.17%	95.67%	91.35%
Client 7	97.40%	91.85%	94.72%	90.96%
Client 8	94.58%	86.79%	87.21%	89.12%
Client 9	96.96%	94.13%	93.68%	93.52%
Client 10	93.47%	91.46%	88.74%	89.91%
Client 11	97.19%	95.29%	96.85%	94.16%
Client 12	95.33%	93.58%	92.43%	92.67%
Client 13	94.91%	88.62%	89.36%	88.94%
Client 14	96.58%	94.81%	95.17%	93.28%
Client 15	93.80%	90.35%	91.82%	90.47%
Avg.	95.59%	92.41%	92.36%	91.79%

**Figure 13:** Comparison of Test Accuracies for different federated learning techniques over the rounds.

transfers during early training stages. The communication efficiency of Cur Fed is particularly valuable in bandwidth-constrained environments, such as IoT networks for human activity recognition, where reducing data transmission is

**Figure 14:** Communication cost comparison of Proposed Technique with other competing techniques.

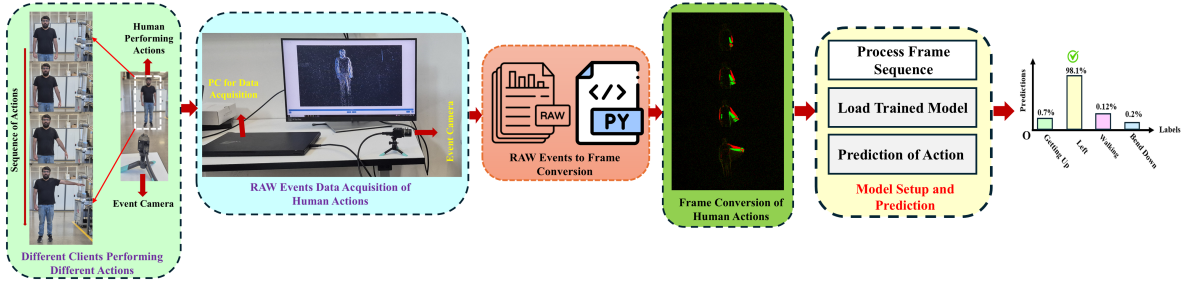
critical for practical deployment. This reduction in communication overhead, combined with the enhanced performance demonstrated in our ablation studies, establishes Curriculum Federated Learning as a more resource-efficient approach for distributed learning on heterogeneous action recognition datasets without compromising model accuracy or generalization capabilities.

6.4. Experimental Validation

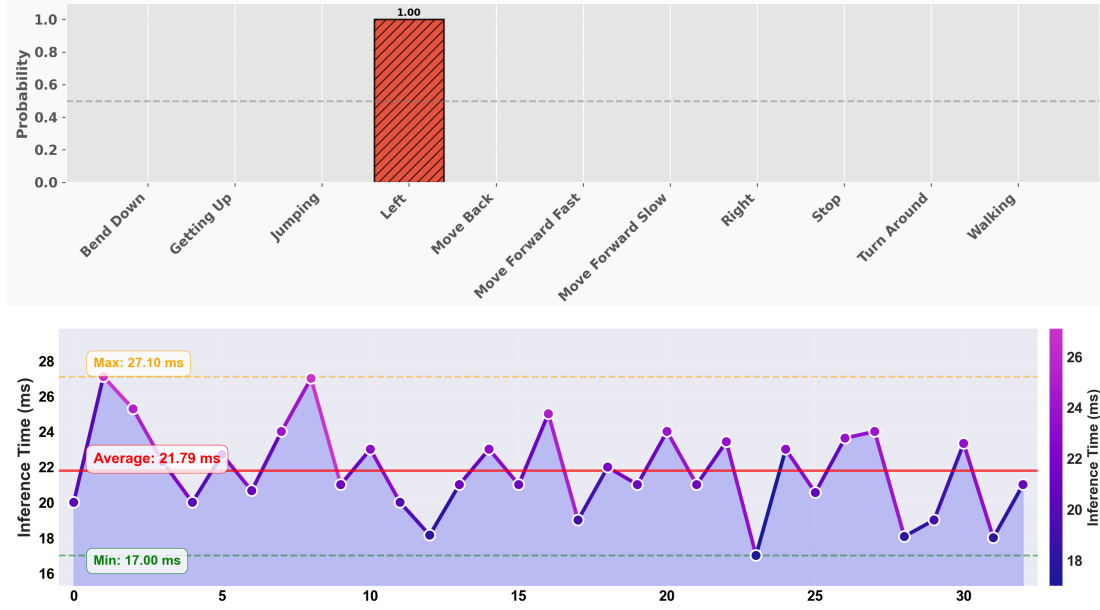
The experimental framework comprises a comprehensive multi-stage pipeline for neuromorphic vision-based human action recognition. Initially, human subjects perform a predefined set of actions while being recorded by an event camera, which captures asynchronous brightness changes rather than conventional frame-based imagery. The raw event data, consisting of pixel coordinates, polarity, and precise timestamps, is acquired through a dedicated PC-based acquisition system. Subsequently, these sparse spatiotemporal events undergo conversion to frame-based representations through custom Python algorithms, transforming the asynchronous data stream into a processable format while preserving the temporal dynamics crucial for action discrimination. The converted frames are further processed through specialized encoding techniques that highlight motion patterns through color-coded visualizations, enhancing the salient features of each action. In the final stage, these processed frames are fed into a trained deep learning model following a sequential workflow of model initialization and inference.

The experimental results presented in the figures provide compelling evidence of the system's performance characteristics. The upper graph illustrates the probabilistic classification outcome across eleven distinct human actions, with the "Left" action demonstrating perfect classification accuracy (probability of 1.00), while all other action classes show negligible false positive rates. This suggests the model exhibits exceptional discriminative capability for the "Left" movement pattern. The lower graph displays the system's computational efficiency through inference time measurements across multiple test iterations, revealing an average processing time of 21.79 ms per frame, with performance ranging between a minimum of 17.00 ms and a maximum

Event Driven Multi-Task



(a) Detailed Experimental Setup for the real time testing of proposed model



(b) Real time prediction and inference time analysis

Figure 15: Experimental setup and results analysis of the proposed model

of 27.10 ms. This temporal analysis indicates the system operates at approximately 45 frames per second on average, making it suitable for real-time applications. The inference time exhibits moderate fluctuations across the 30 test samples, likely attributable to variations in action complexity or system load conditions, but consistently remains within a performance envelope appropriate for real-time human action recognition tasks. These results collectively demonstrate the effectiveness of the event-based neuromorphic approach for accurate and efficient human action recognition.

6.5. Discussion

The proposed Curriculum Federated Learning (Cur_Fed) approach demonstrates significant advantages over traditional federated learning techniques in both performance and efficiency. By strategically integrating clients based on performance thresholds, Cur_Fed achieved the highest average accuracy (94.87%) while maintaining remarkable consistency across all clients, even those with specialized

data distributions. This is in stark contrast to FedAvg, FedProx, and FedCurv, which showed notable weaknesses with specific clients (particularly Client 8) and higher variability in performance. The ablation studies further validate the architectural design choices, with the Global Temporal Attention mechanism (Layer 3) proving essential for effective knowledge transfer across heterogeneous client data, as evidenced by the dramatic 60% reduction in Knowledge Transfer Efficiency when this component was removed. The dynamic training epoch analysis reinforced the importance of sufficient local training, with the 15-epoch configuration demonstrating substantially superior performance throughout all training stages. Perhaps most remarkably, Cur_Fed achieved these performance improvements while simultaneously reducing communication overhead by 27.3% compared to competing approaches. This efficiency stems from the curriculum learning strategy's sequential client integration, which eliminates unnecessary data transfers during early training rounds. The client contribution weighting scheme further optimized knowledge accumulation by

balancing influence based on class representation, allowing data-rich clients to establish strong foundational knowledge while still incorporating specialized expertise from clients with narrower data distributions. These results collectively demonstrate that Curriculum Federated Learning effectively addresses the core challenges of federated learning in non-IID settings by mitigating catastrophic forgetting, maximizing knowledge transfer efficiency, and reducing resource requirements—making it particularly valuable for real-world distributed human action recognition applications where data heterogeneity and communication constraints are significant concerns.

7. Conclusion

This work introduces *CurFed-CHARM*, a novel curriculum federated learning framework for neuromorphic vision-based human action recognition that effectively addresses the challenges of privacy preservation, communication efficiency, and performance with heterogeneous non-IID data. Through systematic experimentation and analysis, we have demonstrated the significant advantages of our approach over traditional federated learning methods. Our results conclusively show that the curriculum-based integration of clients with performance-driven thresholds substantially improves model convergence and generalization capabilities. The proposed 3D CNN architecture with Global Temporal Attention mechanism proves essential for action recognition in federated settings, with ablation studies confirming its critical role in facilitating knowledge transfer across clients with different class distributions. The 60% improvement in Knowledge Transfer Efficiency provided by this mechanism highlights its importance in heterogeneous federated environments. *CurFed-CHARM* achieves an impressive 94.87% average accuracy across all clients on E-FedAct dataset and 95.59% accuracy on PAF dataset while reducing communication costs by 27.3% compared to conventional federated learning approaches. This dual advantage of superior performance and enhanced efficiency makes our framework particularly suitable for resource-constrained edge devices and IoT networks where bandwidth limitations are significant concerns. Furthermore, our analysis of varying local training epochs reveals that sufficient local optimization (15 epochs) before model aggregation is crucial for establishing robust knowledge foundations in curriculum learning. This finding provides valuable insights for implementing efficient federated learning strategies in practical deployment scenarios. *CurFed-CHARM* opens promising research directions for neuromorphic vision-based systems that can operate under privacy constraints with reduced communication overhead. Future work could explore adaptive threshold mechanisms, client selection strategies based on data quality metrics, and extensions to other modalities beyond vision-based action recognition. As privacy-preserving distributed learning continues to gain importance in real-world applications, our curriculum-based approach offers a practical and effective

solution that balances performance, privacy, and efficiency for next-generation intelligent systems.

Acknowledgement

This research is supported by the Artificial Intelligence, Biomechatronics, and Collaborative Robotics research group at the Top Research Center Mechatronics (TRCM), University of Agder (UiA), Norway.

References

- [1] P. Pareek and A. Thakkar, "A survey on video-based human action recognition: Recent updates, datasets, challenges, and applications," *Artificial Intelligence Review*, vol. 54, no. 3, pp. 2259–2322, 2021.
- [2] Y. Zhang, K. Ding, J. Hui, S. Liu, W. Guo, and L. Wang, "Skeleton-rgb integrated highly similar human action prediction in human-robot collaborative assembly," *Robotics and Computer-Integrated Manufacturing*, vol. 86, p. 102659, 2024.
- [3] D. Gehrig and D. Scaramuzza, "Low-latency automotive vision with event cameras," *Nature*, vol. 629, no. 8014, pp. 1034–1040, 2024.
- [4] L. Cultrera, F. Becattini, L. Berlincioni, C. Ferrari, and A. Del Bimbo, "Spatio-temporal transformers for action unit classification with event cameras," *arXiv preprint arXiv:2410.21958*, 2024.
- [5] C. Zhang, Y. Xie, H. Bai, B. Yu, W. Li, and Y. Gao, "A survey on federated learning," *Knowledge-Based Systems*, vol. 216, p. 106775, 2021.
- [6] S. A. Baby, B. Vinod, C. Chinni, and K. Mitra, "Dynamic vision sensors for human activity recognition," in *Proc. of the 4th IEEE IAPR Asian conference on pattern recognition (ACPR)*, 2017, pp. 316–321.
- [7] X. Clady, J.-M. Maro, S. Barré, and R. B. Benosman, "A motion-based feature for event-based pattern recognition," *Frontiers in neuroscience*, vol. 10, p. 594, 2017.
- [8] R. Ghosh, A. Gupta, A. Nakagawa, A. Soares, and N. Thakor, "Spatiotemporal filtering for event-based action recognition," *arXiv preprint arXiv:1903.07067*, 2019.
- [9] S. U. Innocenti, F. Becattini, F. Pernici, and A. Del Bimbo, "Temporal binary representation for event-based action recognition," in *Proc. of the 25th IEEE International Conference on Pattern Recognition (ICPR)*, 2021, pp. 10426–10432.
- [10] G. Layher, T. Brosch, and H. Neumann, "Real-time biologically inspired action recognition from key poses using a neuromorphic architecture," *Frontiers in neurorobotics*, vol. 11, p. 13, 2017.
- [11] A. Amir et al., "A low power, fully event-based gesture recognition system," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 7388–7397. DOI: 10.1109/CVPR.2017.781.
- [12] B. R. Pradhan, Y. Bethi, S. Narayanan, A. Chakraborty, and C. S. Thakur, "N-har: A neuromorphic event-based human activity recognition system using memory surfaces," in *Proc. of the IEEE International Symposium on Circuits and Systems (ISCAS)*, 2019, pp. 1–5.
- [13] C. Liu, X. Qi, E. Y. Lam, and N. Wong, "Fast classification and action recognition with event-based imaging," *IEEE access*, vol. 10, pp. 55638–55649, 2022.
- [14] Q. Liu, D. Xing, H. Tang, D. Ma, and G. Pan, "Event-based action recognition using motion information and spiking neural networks," in *IJCAI*, 2021, pp. 1743–1749.
- [15] S.-C. Yang, M.-L. Wu, and Y.-K. Wang, "Human activity recognition with spiking neural network," in *Proc. of the IEEE 11th Global Conference on Consumer Electronics (GCCE)*, 2022, pp. 482–483.

- [16] X. Lin, M. Liu, and H. Chen, “Spike-har++: An energy-efficient and lightweight parallel spiking transformer for event-based human action recognition,” *Frontiers in Computational Neuroscience*, vol. 18, p. 1508297, 2024.
- [17] G. Chen et al., “Neuromorphic vision-based fall localization in event streams with temporal–spatial attention weighted network,” *IEEE transactions on cybernetics*, vol. 52, no. 9, pp. 9251–9262, 2022.
- [18] S. S. Prasad, N. K. Mehta, H. Kumar, A. Banerjee, S. Saurav, and S. Singh, “Hybrid snn-based privacy-preserving fall detection using neuromorphic sensors,” in *Proc. of the Fourteenth Indian Conference on Computer Vision, Graphics and Image Processing*, 2023, pp. 1–9.
- [19] F. J. Moreno-Rodríguez, V. J. Traver, F. Barranco, M. Dimiccoli, and F. Pla, “Visual event-based egocentric human action recognition,” in *Proc. of the Iberian Conference on Pattern Recognition and Image Analysis*, Springer, 2022, pp. 402–414.
- [20] X. Wang et al., “Hardvs: Revisiting human activity recognition with dynamic vision sensors,” in *Proc. of the AAAI Conference on Artificial Intelligence*, vol. 38, 2024, pp. 5615–5623.
- [21] C. Plizzari et al., “E2 (go) motion: Motion augmented event stream for egocentric action recognition,” in *Proc. of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 19935–19947.
- [22] S. Miao et al., “Neuromorphic vision datasets for pedestrian detection, action recognition, and fall detection,” *Frontiers in neuro-robotics*, vol. 13, p. 38, 2019.