

Application of Deep Learning Method under Federated Learning for Human Action Recognition

Yongpeng Sun

*Glasgow College, University of Electronic Science and Technology of China, Chengdu, China
2023190501026@std.uestc.edu.cn*

Abstract: Human Action Recognition (HAR) is growing in importance in various fields such as security, healthcare, and sports, and deep learning methods such as Convolutional Neural Networks (CNNs) have reached promising results. This study introduces federated learning (FL) techniques to integrate CNN models for multi-device action recognition to improve model performance while ensuring privacy. Specifically, the Residual Network (ResNet) 50 model is used as the backbone for feature extraction and human action classification. Second, FL is used to train classification models across multiple devices. This study aggregates model updates using the Federated Averaging (FedAvg) algorithm. The method uses data augmentation techniques to improve generalization and prevent overfitting. The model is trained on a public HAR dataset from Hugging Face. Experimental results present that the FL-based model achieves competitive performance in action recognition while maintaining data privacy. However, the model also faces the challenge of overfitting due to the significant gap between training and validation accuracies. Future work will explore optimization strategies such as Federalized Proximal (FedProx) and other regularization methods to enhance the model's generalization and address the overfitting problem.

Keywords: Human Action Recognition (HAR), Federated Learning, Convolutional Neural Networks (CNNs).

1. Introduction

With the development of smart devices and advanced applications in artificial intelligence, Human Action Recognition (HAR) based on computer vision has gained significant importance in various fields such as security, sports, and healthcare. HAR analyzes video or image sequences to identify and classify different actions or behaviors automatically. Traditional action recognition methods rely on handcrafted feature extraction and classifier design. However, Convolutional Neural Networks (CNNs) have become mainstream due to their superior feature learning capabilities [1]. As the volume of data continues to grow, privacy concerns and transmission efficiency issues emerge in traditional centralized learning paradigms. Therefore, research has focused on developing efficient action recognition methods while preserving data privacy [2]. Federated learning (FL), an emerging decentralized machine learning approach, enables collaborative model training without data leaving local devices, providing a promising solution to these challenges [3]. While FL ensures data privacy and reduces communication costs, it also introduces new challenges, such as data heterogeneity, communication efficiency, and model convergence in non-independent and identically distributed

environments [4]. Therefore, integrating FL with CNN-based HAR models is a promising yet complex research direction that requires further investigation.

For previous research in action recognition, many studies have focused on utilizing deep learning techniques to improve recognition accuracy and robustness. CNN-based action recognition methods extract spatial and temporal features from video frames, achieving high recognition performance [4]. Despite the effectiveness of CNN in HAR, deploying these models in real-world applications presents practical challenges. As a result, traditional centralized training methods require transferring large-scale data to cloud servers, which imposes a heavy communication burden and raises data privacy concerns. FL has been introduced into HAR as a distributed machine learning framework to address this issue. FL enables training on local devices while transmitting only model updates, thus preventing raw data exposure and effectively preserving user privacy [4]. Several studies have attempted to integrate federated learning with CNNs, demonstrating improved recognition performance in applications such as smart homes and health monitoring systems [5, 6]. McMahan introduced the Federated Averaging (FedAvg) algorithm, significantly reducing communication overhead while maintaining competitive performance. However, most existing approaches focus on single-device environments, and applying more types of FL methods to different situations remains a challenge that requires further exploration.

The primary objective of this study is to explore the application of FL combined with CNNs in multi-device HAR. Specifically, the research first employs a CNN-based deep neural network since the Thesis adopts Residual Network (ResNet) 50 as the backbone network for feature extraction and classification of human actions. To improve generalization, the model is trained on the HAR dataset from Hugging Face using techniques including random cropping, flipping, and color jittering for data augmentation. Second, the research integrates federated learning into the framework, designing an efficient model update and parameter aggregation strategy to enhance collaborative learning across multiple devices while minimizing communication overhead. Third, the thesis applies FedAvg as a federated learning algorithm to analyze its performance in multi-device environments, investigating the trade-offs between model accuracy and computational efficiency compared to the original CNNs. Additionally, this study addresses challenges arising from non-average data distributions and comes up with suggestions for improvement. This approach enables effective action recognition across multiple devices, ensuring high performance and fast response times in real-world applications. Future work will explore optimization strategies like Federated Proximal (FedProx) to resolve these issues and enhance federated learning's applicability in real-world action recognition systems.

2. Methodology

2.1. Dataset description and preprocessing

The HAR dataset from Hugging Face contains 18000 data images, which are composed of 15 types of human action (shown in Figure 1) [7]. Each image corresponds to a specific action label, with the image files stored in a directory and the label information provided in a Comma-Separated Values (CSV) file. Different preprocessing techniques are used for training and validation sets to ensure the dataset is suitable for deep learning model training. Data augmentation is applied to the training dataset to improve generalization and avoid the over-fitting problem. Firstly, a random resized crop is used to randomly resize and crop the images into 224×224 pixels, which can enhance diversity in input scales and compositions. Next, the application of Random Horizontal Flip helps flip images horizontally with a 50% chance, allowing the model to learn actions in different directions and enhancing its robustness to the direction of motion. Moreover, with the introduction of Color jittering, the brightness and contrast are slightly changed to simulate different light conditions. Unlike the training set, resizing and center cropping are used for the test dataset. Lastly, the training and test sets

are normalized for uniformity. These preprocessing steps help minimize noise in the data, improve training effectiveness, and enhance the ability of the model to generalize across diverse inputs.



Figure 1: Human action example of running [7]

2.2. Methods

Based on the techniques used in the project, the model mainly contains two key parts: ResNet50 and Fed-avg. This method benefits from ResNet50's deep learning capabilities for high-level feature extraction while leveraging Federated Learning's privacy-preserving nature to enhance the model's application in real-world environments. The project is composed of five parts, which are shown below in Figure 2.

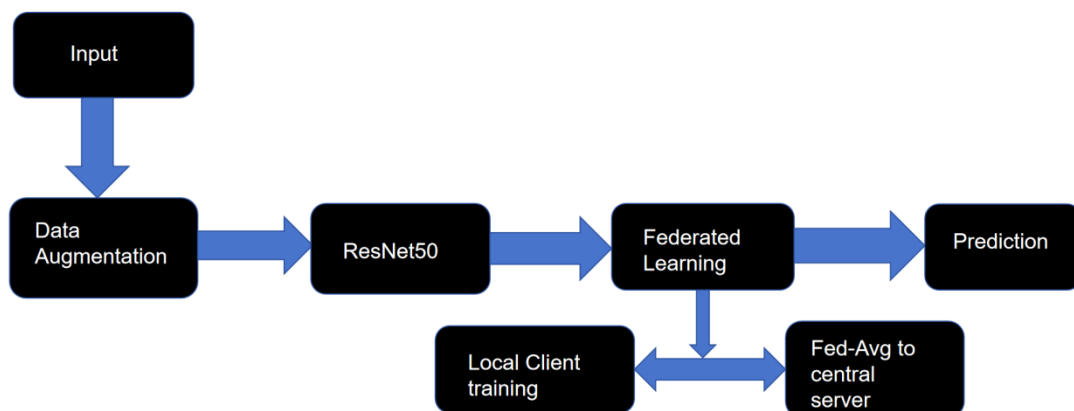


Figure 2: The pipeline of the model (photo credit: original)

2.2.1. ResNet50

The convolutional base of the model is constructed using ResNet50, a widely recognized deep learning architecture known for effectively handling the vanishing and exploding gradient problems. Deep neural networks frequently have these problems because of their ability to learn difficultly for models due to the networks' increasing depth and difficulty in learning. ResNet50 solves this problem by using skip connections, which makes it easier for the network to learn by allowing the gradients to go beyond specific layers. Even as it increases, which aids in maintaining performance during training, these skip connections or residual connections allow the network to retain important information from previous layers. Therefore, it is widely used for feature extraction tasks because ResNet50 can be used to train much deeper networks without experiencing performance degradation.

In this model, ResNet50 is employed as a feature extractor. The model processes the input images, generating feature maps for subsequent classification stages. The input images, originally of size 48x48 pixels, are resized to 224x224 pixels. This resizing ensures that the model captures more detailed information in the images, which is especially important for recognizing complex actions and fine-grained features. The increased image resolution helps retain more spatial details essential

for distinguishing between subtle differences in human actions, which might not be captured in lower-resolution images.

Moreover, ResNet50 is adapted to handle a 15-class classification problem for HAR. The standard ResNet50 model's final fully connected layer is substituted with a custom layer that produces 15 units, each representing one of the action categories in the HAR dataset. This modification allows the model to classify the processed images into 15 distinct human actions, making it suitable for the action recognition task at hand. This approach combines the ability of deep learning with a tailored classification layer to improve the model's performance in recognizing complex human actions.

2.2.2. Federated Learning (FL)

FL is a decentralized machine learning approach that facilitates collaborative model training across multiple devices or clients while maintaining data privacy. Unlike traditional machine learning methods, where raw data is shared with a central server for training, FL allows clients to train models locally on their private datasets. Only updates to the model, like weights, are transmitted to the central server for aggregation, ensuring that the data remains on the local devices. This method ensures that sensitive information, such as personal activity tracking or healthcare data, remains private, making it particularly useful for applications where data security and privacy are paramount.

This approach uses FL to train the ResNet50 model across multiple clients. Each client trains a local model on its private dataset, and after a certain number (20) of local epochs, the model weights are sent to a central server. The server then aggregates these updates using the FedAvg algorithm, which averages the model weights from all clients and updates the global model. The international model is then sent back to the clients for further training. This collaborative learning process allows the model to benefit from diverse data across various clients while maintaining privacy.

This strategy is beneficial when data is distributed among multiple locations, and sharing raw data is impossible due to privacy or regulatory concerns, as shown in sectors including healthcare and finance. By using FL, each client benefits from the global model without exposing their sensitive data to other clients or the central server. Also, the collaborative nature of FL increases the model's generalization capability, since it is exposed to a bigger variety of data, which helps the model learn more robust features across multiple circumstances.

2.2.3. Loss function

In the case of human action recognition, the Cross-Entropy Loss function is used, which is frequently applied to problems with multiple classification classes. This loss function assesses the discrepancy by comparing the expected probabilities for every class to the actual class labels. The model tries to reduce this difference during training, which increases the model's capacity to classify new data precisely. The formula for calculating the Cross-Entropy Loss is as follows:

$$L = - \sum_{i=1}^C y_i \log(\hat{y}_i) \quad (1)$$

where C is the number of classes, y_i is the ground truth, $\hat{y}_i$ is the predicted probability for class i . The optimizer used for minimizing the loss is AdamW, an improved version of the Adam optimizer that includes weight decay regularization. AdamW is particularly effective in training deep networks by adjusting the learning rate based on both the gradients and the moving averages of the past gradients. This helps in achieving faster convergence and better generalization.

2.3. Details

Several key aspects of the implementation of the proposed model are highlighted. The learning rate is set to 0.0001 and decreases by a factor of 10 when the validation accuracy does not change

anymore. A batch size of 16 is used, and the model trains for 20 epochs. The optimizer is AdamW, chosen for its efficient gradient descent handling with weight decay [8]. Data augmentation techniques, including random resized cropping, horizontal flipping, and color jittering (brightness and contrast adjustments), are applied to increase the dataset diversity and reduce overfitting. These techniques expose the model to different perspectives and scales of human actions, improving its generalization ability. The dataset consists of RGB images where human actions are centered, and the background remains relatively uniform, allowing the model to focus on the action features for classification.

3. Discussion

As shown in Figure 3, the confusion matrix shows the performance of the model after 20 rounds of training. Each row of the matrix corresponds to the actual label, and each column represents the predicted label. The intensity of the blue color in each cell reflects the count of samples classified into that category. Higher values are represented by darker blue cells, indicating better classification accuracy for those classes. The diagonal values of the matrix indicate the samples that have been correctly classified. Most of these values are relatively high, indicating that the model performs well classifying most action categories. For example, the class corresponding to "Running" (row 10, column 10) has a high value of 148, showing that the model is effectively identifying this action.

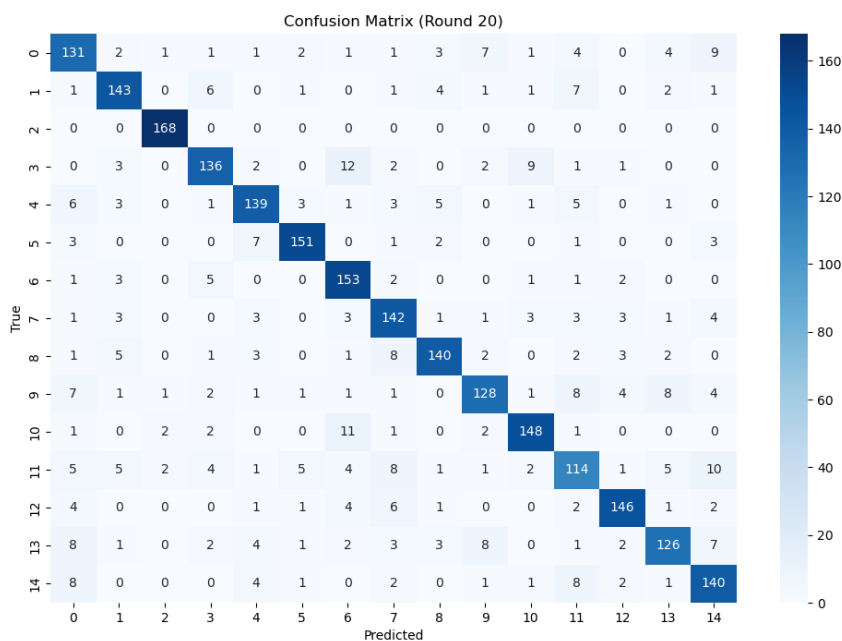


Figure 3: Confusion matrix of round 20 (photo credit: original)

The left graph in Figure 4 also shows that training loss decreases steadily, which clearly indicates the model is learning and improving its predictions. This decrease in training loss means the model correctly fits the training data. In Figure 4, the training accuracy and validation accuracy are shown in the right graph. As it shows, the training accuracy is increasing with each period. The accuracy reaches a high value of 85% at the conclusion of training. The validation accuracy increases; however, at a slower rate than the training accuracy. Initially, the validation accuracy lags behind the training accuracy, but it eventually stabilizes as the model reaches a point of convergence. The validation

accuracy stabilizing at a certain value implies that although the model learns from the training data, its capacity to generalize to unseen data is constrained. The difference between the training and validation accuracy is a sign of overfitting, where the model performs well on the training data but faces challenges in generalizing to the validation data.

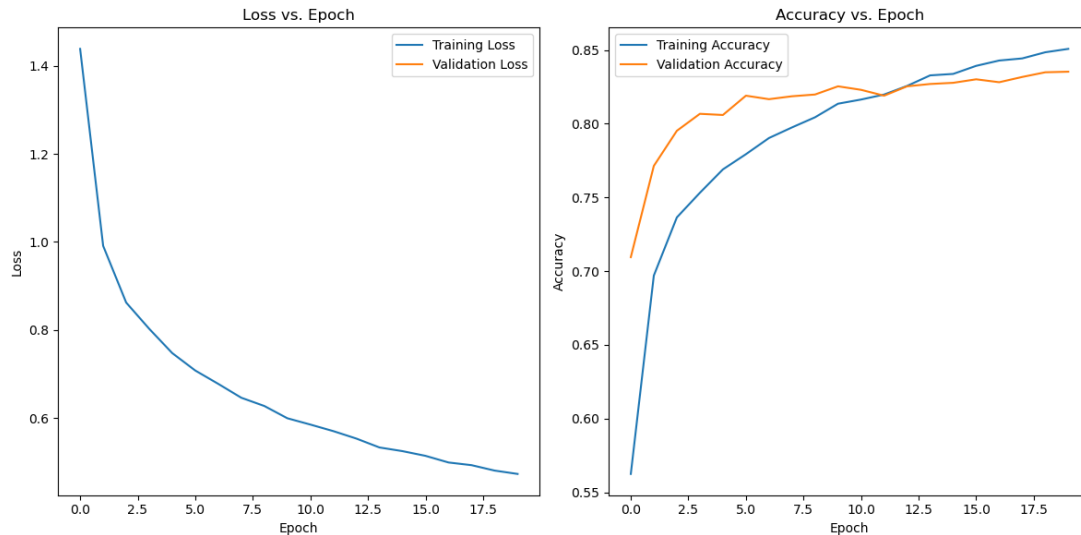


Figure 4: Loss and accuracy function of round 20 (photo credit: original)

According to the results performed by the model, it is the typical behavior of a deep learning model during training. The decrease in training loss and the corresponding increase in training accuracy indicate that the model is learning effectively. However, the gap between training and validation accuracy suggests that overfitting is beginning to occur. The results indicate that the model becomes overly focused on the training data, leading to poorer performance on new or unseen data. In the future, introducing regularization techniques such as dropout and early stopping may help overcome the difficulty of overfitting. Meanwhile, some human actions may have similar gestures, leading to mistakes. Improving the feature extraction capabilities of the model or using more diverse training data could help reduce such errors. From the analysis, the dataset is not sufficiently augmented; the model might memorize specific training data features, reducing its ability to generalize to new data. However, increasing data diversity through augmentation could help the model learn more robust features. While the model shows good learning progress, the slight overfitting and gap between training and validation performance highlight areas for further improvement. Implementing techniques to reduce overfitting and enhance generalization could further optimize the model's performance.

4. Conclusion

This research introduces a new strategy to HAR that uses FL and CNN to protect data confidentiality when increasing model generalization among several devices. The method uses ResNet50 as the leading network for feature extraction and classification. FL permits a teamwork model training without sharing raw data. The FedAvg algorithm is used to aggregate model updates between devices. The results suggest that adding FL to the HAR model might successfully decrease data transfer cost while guaranteeing data privacy. Experiments on the HAR dataset show that the model significantly increases the HAR accuracy relative to the centralized model. However, overfitting remains difficult, as demonstrated by the difference between training and validation accuracy. This indicates that whereas the model learns well on the training set, its ability to generalize to new data needs additional

improvement. In the future, methods such as FedProx and other regularization methods will be studied to solve the overfitting problem and increase the model's generalization ability.

References

- [1] Wang, Z., Lu, H., Jin, J., et al. (2022). Human action recognition based on improved two-stream convolution network. *Applied Sciences*, 12(12), 5784.
- [2] Sun, Y., Ochiai, H., Esaki, H. (2021). Decentralized deep learning for multi-access edge computing: A survey on communication efficiency and trustworthiness. *IEEE Transactions on Artificial Intelligence*, 3(6), 963-972.
- [3] Chen, M., Shlezinger, N., Poor, H. V., et al. (2021). Communication-efficient federated learning. *Proceedings of the National Academy of Sciences*, 118(17), e2024789118.
- [4] Tran, D., Bourdev, L., Fergus, R., Torresani, L., & Paluri, M. (2015). Learning spatiotemporal features with 3D convolutional networks. *Proceedings of the IEEE International Conference on Computer Vision*, 4489-4497.
- [5] Chen, Y., Qin, Z., Wang, J., Yu, C., & Cheng, X. (2022). FedHAR: Federated learning for human activity recognition with wearable sensors. *IEEE Internet of Things Journal*, 9(2), 899-910.
- [6] Zhao, Y., Song, Y., Song, J., Xu, Y., Sun, L., & Zhou, W. (2021). Federated learning for human activity recognition: A privacy-preserving approach. *IEEE Transactions on Mobile Computing*.
- [7] Kaggle dataset. (2014). Retried from https://huggingface.co/datasets/Bingsu/Human_Action_Recognition
- [8] Bjorck, J., Weinberger, K. Q., Gomes, C. (2021). Understanding decoupled and early weight decay. *Proceedings of the AAAI Conference on Artificial Intelligence*. 35(8), 6777-6785.