International Journal of Computational Intelligence Systems and Applications. Volume 13, Issue 1, (2026) 16-25
http://www.cayley-nielson.com/ijcisa/

Research on Human Action Recognition Based on Intelligent Optimization Algorithms

Yan Yuhao*[1], Qu Junyang[2], Shuai Jicheng[3], Guo Chang[4], Kang Chenqing[5], Zhang Yezi[6]
[1,2,3,4,5,6]Xi'an Technological University, Xi'an, China
*Corresponding author
E-mail: [1]*2686720942@qq.com*, [2]*2920155808@qq.com,*[3]*1283002906@qq.com,* [4]*2101527206@qq.com,* [5]*2376915397@qq.com,*[6]*2036023112@qq.com*

Abstract

With the advancement of technology, human action recognition techniques are rapidly evolving in scenarios such as intelligent surveillance, virtual reality, smart homes, and health management. As a core direction in this field, deep learning-based human action recognition has demonstrated significant application value in interactive human-robot collaboration and sports health monitoring. This study constructs an intelligent action recognition system for embedded scenarios, following a technical pipeline of "hardware perception - intelligent optimization - accurate classification" to achieve efficient recognition of human actions. Human action images are captured using an STM32-based hardware platform and a CMOS vision sensor, followed by the extraction of 25-dimensional skeletal keypoint sequences via OpenPose. To address the issues of skeletal feature redundancy and model parameter sensitivity, the Particle Swarm Optimization (PSO) algorithm is introduced to optimize feature subset selection and hyperparameter tuning of the LSTM+CNN network. Experimental results show that the optimized model achieves an average recognition accuracy of 95.8% for five actions (standing, waving, walking, raising hands, and squatting), which is 11.5% higher than that of the traditional SVM approach. Moreover, the overall system latency is controlled within 120 ms, satisfying the real-time and low-power requirements of embedded scenarios.

Keywords: Action Recognition, Openpose, Particle Swarm Optimization Algorithm, Long Short-Term Memory Network.

1. Introduction

Human action recognition primarily relies on the spatiotemporal modeling of human postures and motion trajectories. Traditional methods describe action patterns by manually designing features (e.g., optical flow, joint coordinates), while deep learning approaches automatically extract spatial features using convolutional neural networks (CNNs) and model temporal dynamics with long short-term memory (LSTM) networks, enabling end-to-end action recognition [1]. In recent years, skeleton keypoint detection techniques based on pose estimation algorithms such as OpenPose have further improved the robustness of action representation, making action recognition feasible in complex scenarios. Conventional action recognition methods typically follow a paradigm of feature extraction followed by classification. Based on different feature extraction strategies, they can be categorized into local-feature-based and global-feature-based methods. Although these traditional approaches perform well under specific conditions, they exhibit notable limitations: feature design heavily relies on domain knowledge, resulting in limited generalization

ability; they are sensitive to interference such as illumination changes and occlusions; they incur high computational complexity, making real-time processing difficult; and their recognition accuracy in complex scenarios generally falls below 60% [2].

In recent years, significant progress has been made in CNN-based action recognition. Wang Lixin et al. proposed a Spatiotemporal Attention Convolutional Network (STACN), which introduces a channel-spatiotemporal attention mechanism and achieves 95.3% accuracy on the UCF101 dataset [3]. Zhang Hua et al. developed a lightweight 3D convolutional network (Lite3D) that reduces the number of model parameters by 70% while maintaining over 90% recognition accuracy through depthwise separable convolutions and channel pruning [4]. Addressing real-time requirements, Chen Ming et al. proposed an Efficient Video Understanding Network (EV-Net), which adopts a spatiotemporal feature decoupling strategy to achieve a processing speed of 120 frames per second on mobile devices [5]. Li Qiang, in a study published in the Chinese Journal of Computers, noted that the recognition accuracy of current 3D CNN models in complex scenarios still has room for improvement [6]. In terms of temporal modeling, Liu Wei et al. proposed a Multi-scale Spatiotemporal LSTM (MST-LSTM), which achieves 92.1% cross-view accuracy on the NTU RGB+D 120 dataset through hierarchical temporal modeling and feature fusion [7]. Addressing long-sequence modeling, Zhao Gang, in a study published in Acta Automatica Sinica, proposed a Gated Adaptive Memory Network (GAMN) that significantly improves recognition performance on long-term action sequences via a dynamic memory update mechanism [8]. To reduce computational complexity, Wang Jing et al. proposed a Lightweight Temporal Network (LTN), which employs group convolution and knowledge distillation to achieve a threefold inference speedup while maintaining over 90% accuracy [9]. Zhang Min, in a study published in the Journal of Image and Graphics, showed that LSTM networks still suffer from overfitting in small-sample action recognition tasks [10].

Despite the remarkable progress made by deep learning methods, several challenges remain: high computational cost, large parameter volumes in models such as 3D CNNs, reliance on empirical experience for hyperparameter optimization without a systematic methodology, limited generalization ability in small-sample scenarios, and difficulty in meeting real-time requirements for mobile applications [11]. To address the above issues, this paper proposes an action recognition model training scheme based on human skeletal keypoints and lightweight networks.

2. Human Motion Recognition Based on Computer Vision

2.1. Skeleton Feature Extraction Based on OpenPose

The support of the OpenPose model for multi-person pose estimation serves as a key enabler for the algorithmic design of this study. However, the raw skeleton features extracted by OpenPose consist of 25-dimensional joint coordinates, which contain spatial redundancy. Directly feeding these features into conventional models tends to result in low computational efficiency. Moreover, a network structure with fixed hyperparameters struggles to adapt to the varying feature complexity of different actions. To address these issues, this study constructs a lightweight network architecture following the pipeline of "feature extraction - feature optimization- model construction". As illustrated in Figure 1, the proposed architecture first employs OpenPose to extract sequences of skeletal keypoints, forming a base feature set. Subsequently, a particle swarm optimization algorithm is used to dynamically select critical features, thereby mitigating the impact of redundant dimensions on computational speed. Finally, a convolutional network compatible with general-purpose computing power is designed, which, together with intelligent optimization, enables fast feature mapping and classification decision-making.

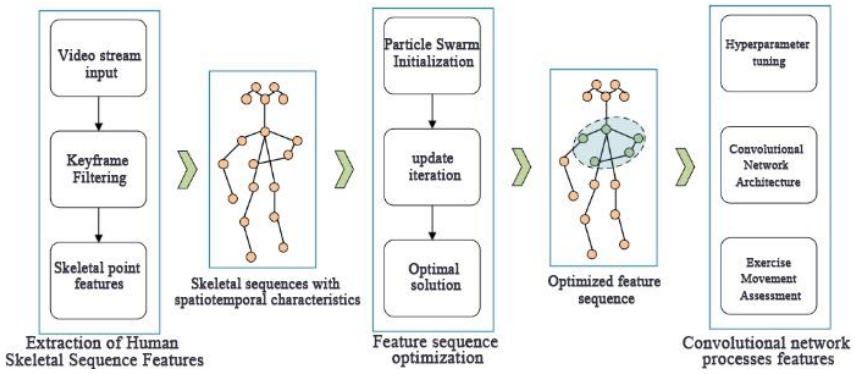

Fig.1. Human Motion Action Recognition Based on OpenPose

2.1.1. Extraction of Human Skeletal Feature Sequences

As a bottom-up recognition algorithm, OpenPose involves a multi-stage, dual-branch convolutional network with substantial computational overhead, resulting in a long inference time. Even when using GPU inference on videos with a resolution of 160×160, the average inference time per frame approaches 50-100 ms. However, real-time captured video typically runs at 30 frames per second. Consequently, when processing various human motion videos, the algorithm may suffer from missing information due to incomplete action postures or excessively fast motion completion, leading to inaccurate recognition results. The human skeletal feature sequence extraction method proposed in this study, which takes OpenPose as the core, can largely resolve this issue. The specific workflow of the method is illustrated in Figure 2.

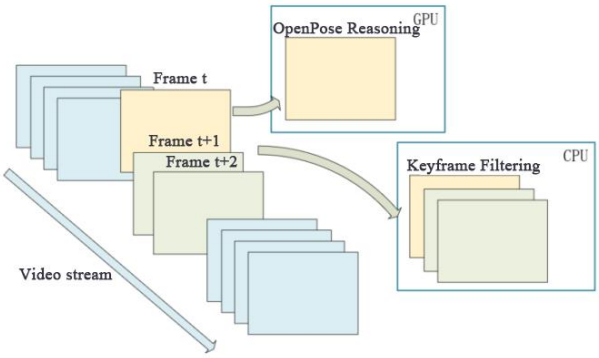

Fig.2. Human Skeletal Sequence Extraction

2.1.2. Key Frame Selection and Inference for Human Motion

Due to the inference time constraints of OpenPose, at least two new image frames are input during the inference process. To maximize the acquisition of human motion information, the approach adopted in this section is to prioritize the extraction of frames with higher motion information content, based on the amount of motion information contained in subsequent frames.

On the basis of completing key frame extraction, the OpenPose-based skeletal keypoint extraction method is performed. The main functional steps of this thread are shown in Figure 3.

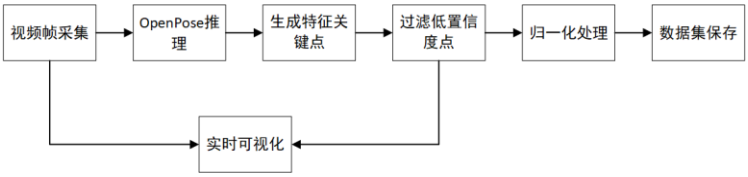

Fig.3. OpenPose Thread Processing Workflow

2.2. Construction of an LSTM Network Based on Particle Swarm Optimization Algorithm

After completing the extraction of human skeletal feature sequences, the next step is to build the training model and start the training process. However, since this chapter only addresses the recognition of simple human motions, the feature sequences extracted by the OpenPose model are somewhat redundant. Such redundancy does not benefit model training; on the contrary, the extra features act as noise, reducing the efficiency of model training. Moreover, the final trained model will suffer from significantly lower recognition accuracy due to this noise. Therefore, this section introduces the Particle Swarm Optimization (PSO) algorithm to optimize the feature sequences. As shown in Figure 4, the PSO algorithm can select the optimal combination of feature vectors from a set of randomly combined feature vectors.

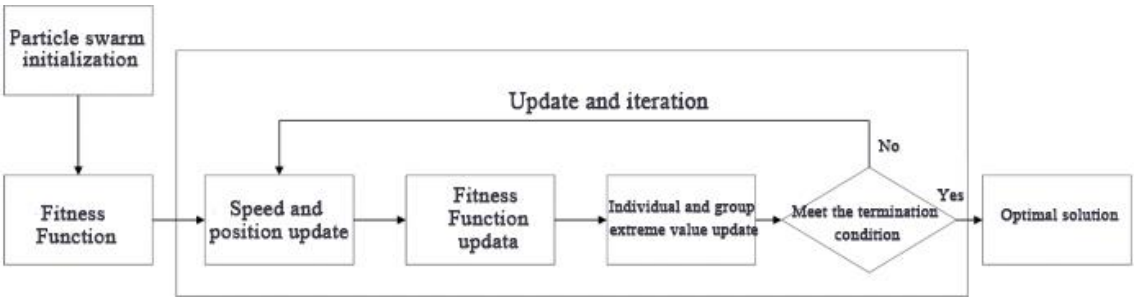

Fig. 4. Particle Swarm Optimization Algorithm

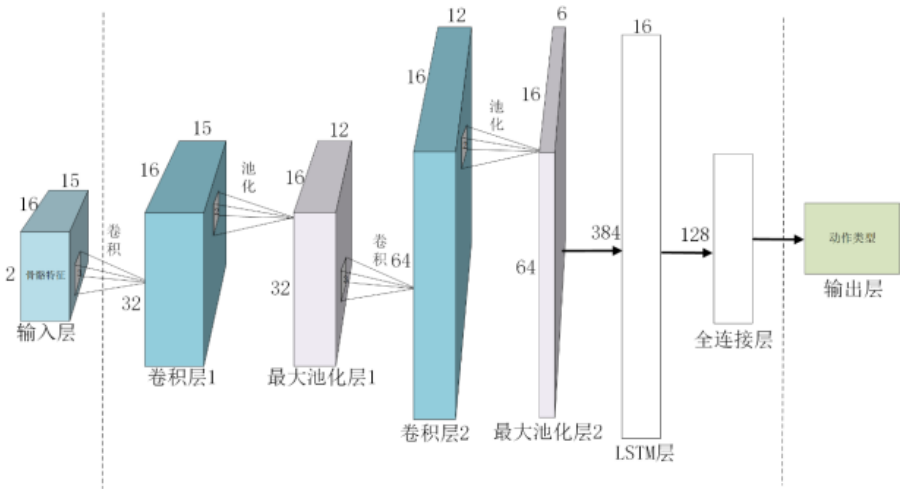

Fig.5. LSTM-CNN Network Architecture

2.2.1. Construction of the Human Motion Action Recognition Network

Using the Particle Swarm Optimization (PSO) algorithm, the screening of temporal feature sequences has been completed, retaining 16 time steps and the coordinate dimensions of 15 joint points. Moreover, the key concepts of this paper are small volume, lightweight design, and simple actions. Therefore, a lightweight LSTM–CNN hybrid scheme is adopted for the network architecture, employing a 'spatiotemporal feature decoupling modeling' strategy to reduce computational complexity. As shown in Figure 5, this architecture, while ensuring recognition accuracy, simplifies the number of layers and parameter scale to accommodate small datasets and edge device deployment requirements. This study will focus on constructing a lightweight action recognition network based on this design.

2.2.2 Hierarchical Architecture of the Human Action Recognition Network

The input layer directly receives the temporal feature sequences filtered by the particle swarm optimization algorithm, containing no redundant dimensions. The input shape is (16, 15, 2), which serves as the input to the subsequent convolutional layers. Tables 1 and 2 present the parameters of the convolutional and pooling layers, respectively, within the hierarchical architecture of the LSTM-CNN network used in this study.

Table 1. Convolutional Layer Parameters

Layer	Number of Filters	Filter Size	Activation Function	Output Shape
Conv Layer 1	32	3	ReLU	(16, 15, 32)
Conv Layer 2	64	3	ReLU	(16, 12, 64)

Table 2. Pooling Layer Parameters

Layer	Pooling Type	Pool Size	Output Shape
Pooling Layer 1	Max Pooling	2	(16, 12, 32)
Pooling Layer 2	Max Pooling	2	(16, 6, 64)

Following the convolutional and pooling operations described above, the first step of the LSTM layer processing is to flatten the feature map of shape (16, 6, 64) output by the convolutional network into a temporal feature sequence of shape (16, 384). This is then used as input to the LSTM layer with the shape (None, 16, 384), where None denotes the batch size. Subsequently, within the LSTM layer, the main computational steps—including forget gate calculation, input gate calculation, cell state update, output gate calculation, and hidden state update—are performed sequentially. Finally, a temporal feature vector of shape (None, 128) is output, passed through a regularization layer that randomly drops 30% of the neurons, and then fed into a fully connected layer to map the temporal feature vector to action categories, thereby completing action classification. Table 3 lists the key parameter settings of the LSTM layer.

Table 3.LSTM Layer Parameters

Units	128	Number of LSTM neurons
Dropout	0.2	Dropout rate for input and forget gate neurons
Return_sequences	False	Returns only the last hidden state
Recurrent_dropout	0.1	Dropout rate for recurrent connections
Units	128	Number of LSTM neurons

To balance convergence speed and stability, the initial learning rate is set to *lr*=0.001. To stabilize gradient variance and accelerate convergence, the momentum parameters are set to β_1=0.9 and β_2=0.999. Finally, an L2 regularization term with λ=0.0001 is added to the fully connected layer to prevent overfitting.

To avoid premature convergence to a local optimum, a cosine annealing learning rate decay is adopted. The corresponding formula is shown in Eq.1, where t denotes the current epoch, *T* the total number of epochs, *lrmax*=0.001, and *lrmin*=0.00001.

$$\mathrm{lr}(t) = lr_{\min} + \frac{1}{2}(lr_{\max} - lr_{\min})(1+\cos(\frac{t\pi}{T})) \tag{1}$$

After implementing the optimization described in this section, although the learning rate decreases slowly in the later stage of training, the model's ability to perform fine-grained search in the feature space is enhanced.

4. Experimental Validation and Result Analysis

Figure 6 illustrates the process of capturing human body images using the camera. Figure 7 shows, respectively, the debugging information printed by the lower computer program through the serial port, the terminal debugging information of the upper computer program, and the real-time transmitted video feed.

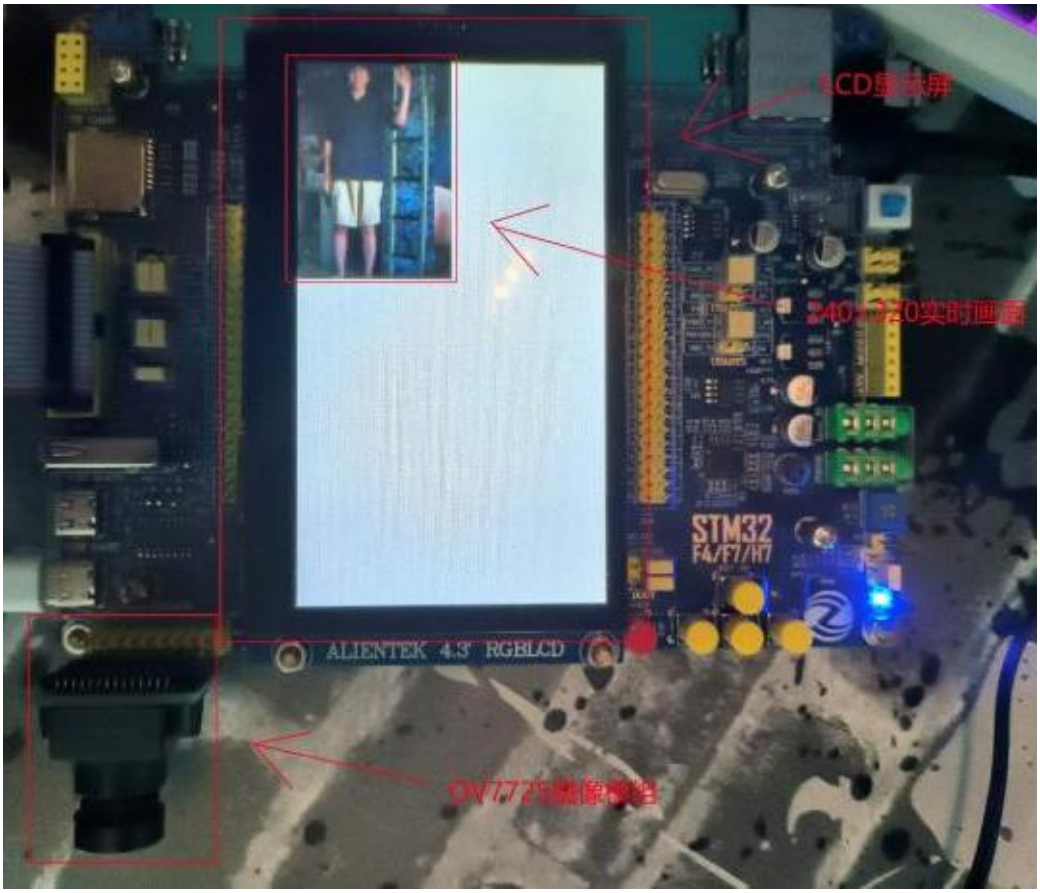

Fig.6. Acquisition of Human Body Images Using a Camera

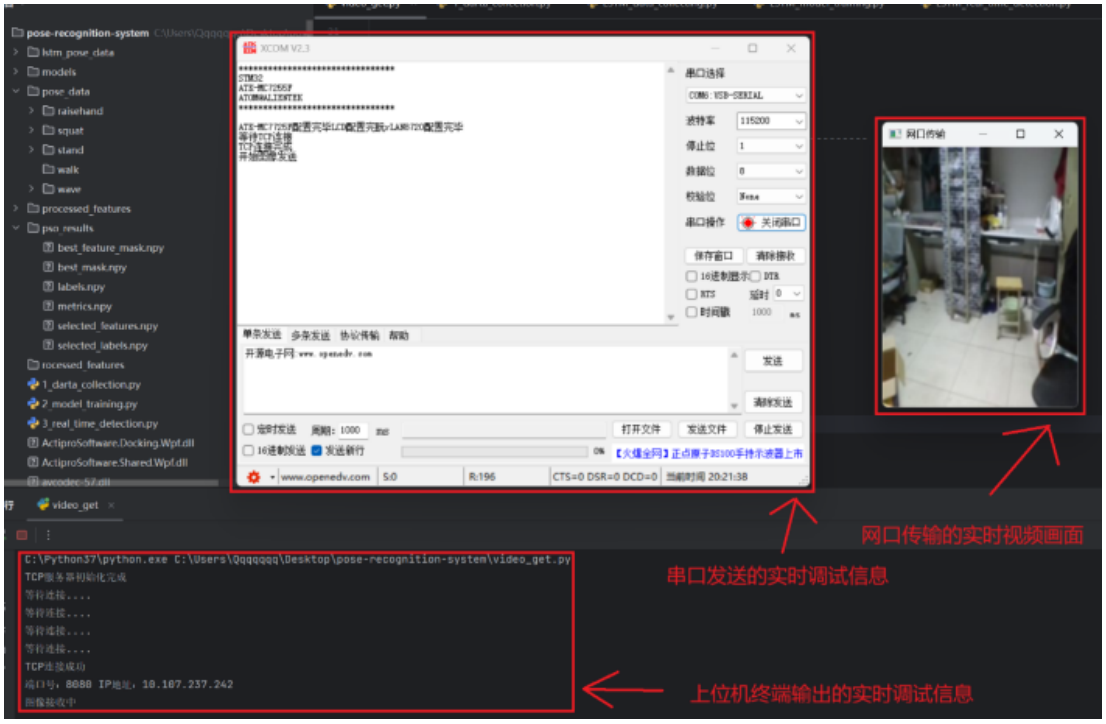

Fig.7. Uploading Images to the Host Computer via Network Communication

The method of collecting human skeletal keypoints using the OpenPose model is feasible. The dataset and test set for each action have been successfully collected (as shown in Figure 8). The PSO algorithm significantly reduces the feature dimensions and training time required for model training without compromising the model's recognition accuracy. The process of feature sequence optimization using the PSO algorithm is illustrated in Figure 9.

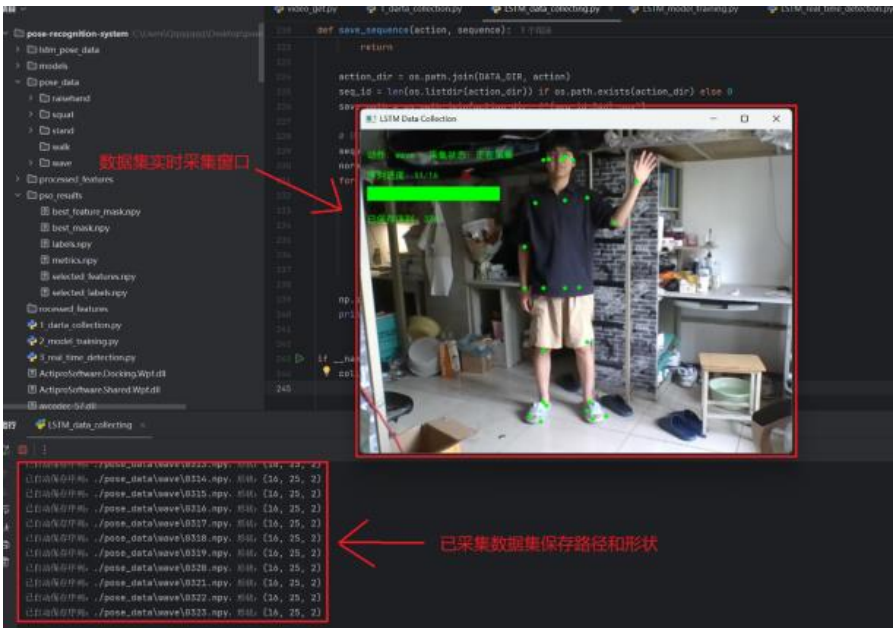

Fig.8. Collection of Human Skeletal Keypoints Using OpenPose

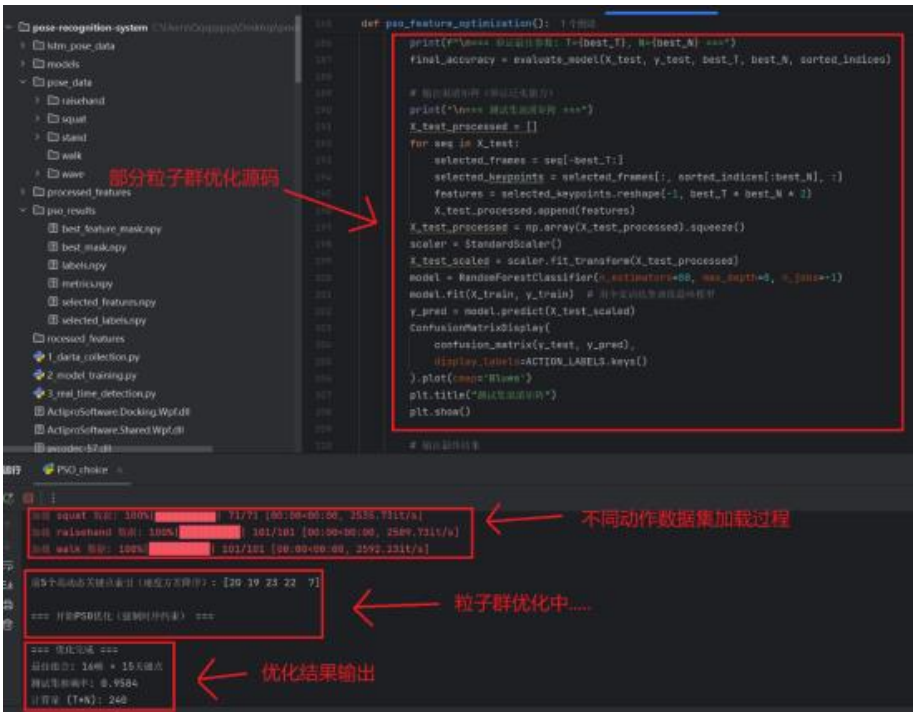

Fig.9. Feature Sequence Optimization Process Using PSO

Figures 10 and 11 show the recognition results of human motion actions (waving and walking), respectively.

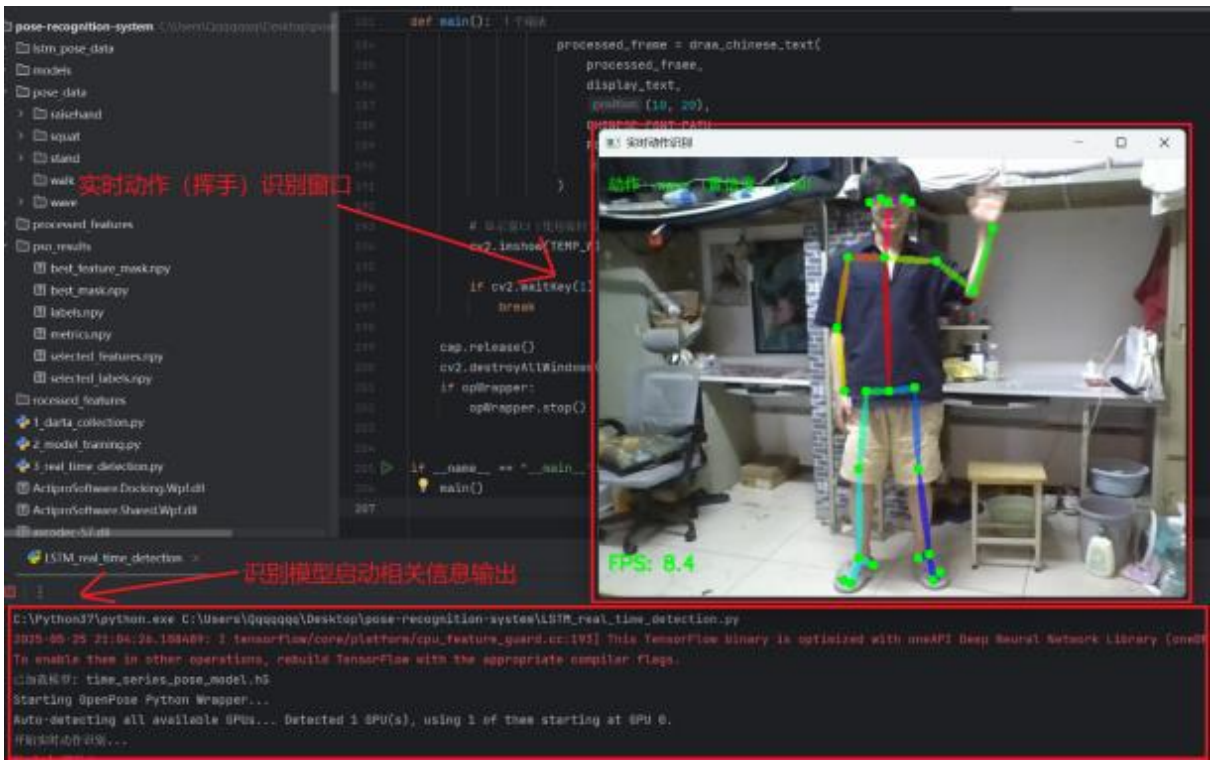

Fig.10. Recognition Effect of the Waving Action

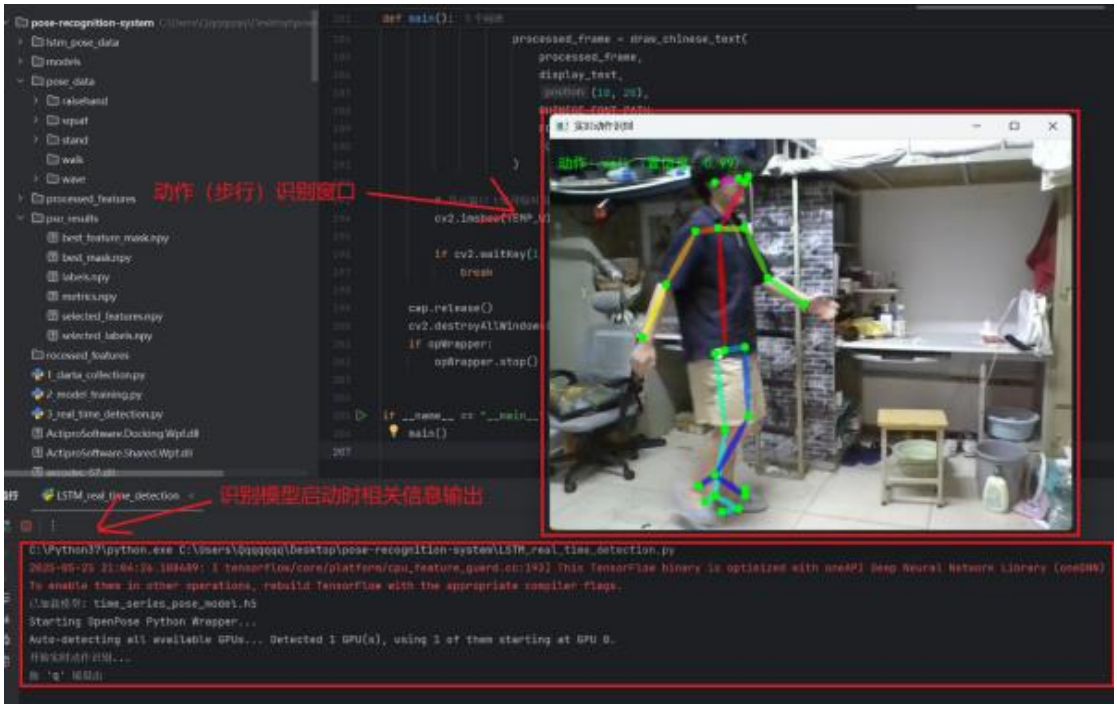

Fig.11. Recognition Effect of the Walking Action

As shown in Figures 10 and 11, the LSTM+CNN-based motion action recognition model achieves high recognition accuracy for simple actions such as standing, squatting, and waving. Moreover, after introducing the PSO algorithm to optimize the feature sequences for specific motion actions, not only does the recognition accuracy for fixed actions increase slightly, but the model training time is also significantly reduced.

5. Acknowledgment

This work was supported by the Training Program of Innovation and Entrepreneurship for Undergraduates. Grant No.S202410702111 and S202410702104. The authors would like to express their sincere gratitude to the funders for their generous financial support. Special thanks also go to all members of the research team for their assistance in data collection and analysis.

References

[1] Zhang J., Sun M.Y. “Application of Machine Learning Algorithms: A Case Study of Video Content Analysis and Video Editing.” Home Theater Technology, 18(2024): 51

[2] Guo H.S. “A Survey of Object Detection: From Traditional Methods to Deep Learning.” Emerging Science and Technology Trends, 3.02(2024): 128.

[3] Lei J.Y., Liang J., Xia M., ZHANG H., TIAN Z.H. “Research on Improved ST-GCN Human Action Recognition Method Fusing Spatiotemporal Attention.” Journal of South-Central Minzu University (Natural Science Edition), 44.04(2025): 526.

[4] Jiang S. “Research on Lightweight Target Tracking Algorithm for Embedded Terminals [D].” Beijing: Beijing University of Posts and Telecommunications, 2022.

[5] Liu X.X., Kuang L.Q., Wang S. “Contrastive Learning for Skeleton-Based Action Recognition with Multi-Scale Spatiotemporal Decoupling.” *Journal of Computer Applications*, 2025,1-10

[6] Wei W., Zheng C., Tang Y. “Skeleton-Based Action Recognition Integrating Spatiotemporal and Motion Information.” *Application Research of Computers*, 2025,1-7

[7] Jin C., Liao N., Chen Y.R. “3D Human Pose Estimation Combining Deep Learning and LBP Texture Features.”*Computer Simulation*, 2025, 42(04): 473.

[8] Xiang L.H. Research on Action Recognition Algorithm Based on Human Skeleton Extraction and Hybrid Data Augmentation [D]. *Hefei: University of Science and Technology of China*, 2024.

[9] Zhang Z.C., Cai Y.C., Zhang L.W. “Process Action Recognition and Analysis Based on Skeleton Sequences.”*Industrial Engineering Journal*, 27(05),2024: 73-80.

[10] Hu Z.P., Wang Y.L., Zhang Q.M. “Small Sample Classification Algorithm for Spatiotemporal Joint Alignment of Skeletal Motion Descriptors in Video Actions.”*Journal of Signal Processing*, 40(08),2024, : 1556.

[11] Zhao S.E., Gong D.Y., Tian Z.S. “Object Detection Algorithm for Complex Traffic Scenarios Based on Improved YOLOv8 Model.”*Journal of Chongqing Jiaotong University (Natural Science Edition)*, 1-9