Forest Chemicals Review
www.forestchemicalsreview.com
ISSN: 1520-0191
July-August 2022 Page No. 339-350
Article History: Received: 30 March 2022, Revised: 8 April 2022, Accepted: 15 April 2022, Publication: 30 April 2022

Lightweight OpenPose Based Body Posture Estimation for Badminton Players

Hui Xiang
College of Physical Education, Southwest Forestry University, Kunming Yunnan 650224, China

Abstract:

In this paper, we optimize the bottom-up human pose estimation network OpenPose in the context of badminton player's body pose, replace the VGG19 backbone network in this model with a lightweight network MobileNet with less number of parameters to achieve a lightweight model, and introduce a polarized self-attention mechanism (Polarized Self-Attention (PSA) is introduced at the front and back ends of the MobileNet backbone network to achieve a reasonable overhead while keeping the input features as high resolution as possible. The results show that the mAP0.5 (%) and mAP0.75 (%) of the optimized model are 85.79% and 70.57%, respectively. Compared with the original OpenPose model, the detection speed is significantly improved, and the FPS of the optimized model is improved by 64.28%, although the average accuracy is slightly reduced. Finally, the experiments show that the optimized lightweight OpenPose model has good estimation accuracy and estimation speed in estimating the body pose of badminton players and can be applied in embedded devices.

Keywords: *OpenPose, Badminton, Human Pose Estimation, MobileNet, Deep Separable Convolution, Polarized Self-attention Mechanism*

I. INTRODUCTION

In recent years, with the rapid development of computer vision [1], human pose estimation (Pose Estimation) [2] is playing an increasing role in people's daily life as a popular direction in the field of computer vision. Human pose estimation can be used in different fields, such as motion recognition, human-computer interaction, and motion capture [3]. Pose estimation also has a wide range of applications in the sports industry, used in sports video analysis [4], where sports events can be analyzed and modeled to provide a comprehensive evaluation of the athletes' performance. In sports, standard sports posture [5] not only has a significant effect on the improvement of athletes' skills, but also can, to a large extent, keep athletes from being injured during sports.

Human pose estimation can be understood as the identification and localization of the position of the body's joint points (head, right hand, left foot, etc.). The methods for human pose estimation are mainly divided into traditional methods and deep learning methods. Traditional methods such as: based on Pictorial Structures, DPM [6-7], etc. Although the traditional methods are efficient in execution time, they cannot make full use of image information, and when the human pose changes a lot, the traditional algorithms cannot accurately portray this deformation, resulting in the same data pose estimation results are not unique, so the scope of application of traditional methods is greatly limited. With the development of computer vision, most of the mainstream pose estimation methods are based on deep learning [2], and the classical ones are AlphapPose, a regional multi-person pose estimation framework proposed by Lu Cewu's team at Shanghai Jiaotong University and Tencent Youtu [8], and OpenPose, an open source framework built based on convolutional neural network proposed by Carnegie Mellon University (CMU) [9]. According to the processing hierarchy, the methods for multi-person pose estimation are divided into "Top-down" and "Bottom-up" detection methods. The top-down approach first detects each person with a target detection frame and then performs single-person pose estimation for each frame, which is highly dependent on the superior performance of the target detection algorithm and has a high accuracy rate of pose estimation, but is time-consuming and prone to pose redundancy. Yang [10] et al. constructed a sequential multiscale feature fusion pose estimation method for the problem of drastic scale changes, large deformation and severe occlusion of human nodes. Zhang [11] et al. applied the human pose estimation method to a medical robot to assist athletes in rehabilitation training. The bottom-up detection method first estimates all the joints at once, then estimates the connection relationship between the joints or the attribution of the joints, and finally connects the joints belonging to the same person to achieve multi-person pose estimation, which is faster, as it does not rely on the target detector for human frame detection, but there is the phenomenon of joint aggregation error, and when multiple people are close together, it is easy to cause the ambiguity problem of nodal point attribution. Li [12] et al. improved the OpenPose model to achieve a lightweight model, and the results showed that their method works well and can operate properly in embedded devices. Liu [13] et al. used the OpenPose human pose estimation algorithm to identify traditional martial arts movements and compare them with standard martial arts movements, which can provide some reference basis for martial arts competitions and martial arts enthusiasts.

The standardization of badminton players' movements and the flexibility of their maneuvers and footwork in the game are key factors in determining the athletes' victory [14]. In order to make the stance estimation algorithm run efficiently in embedded devices, it is necessary to develop network structures with faster detection speed. OpenPose as a bottom-up detection algorithm has faster detection speed, but its number of parameters is too much, in order to further improve the detection efficiency, so the text proposes a lightweight OpenPose based method for estimating the body stance of badminton players.

II. OPENPOSE BENCHMARK MODEL

OpenPose is a bottom-up detection algorithm that relies on convolutional neural network (CNN) and supervised learning for human pose estimation, and its original network architecture is shown in Figure 1. Each module has the same structure and function, and then divided into two branches (branches), one branch generates the heat map of key points (Part Confidence Map, PCM), which is used to characterize the location of key points; the other branch generates Part Affinity Fields (PAF). Each stage of PCM and PAF is solved for loss function (loss), and the final total loss is the sum of all losses.

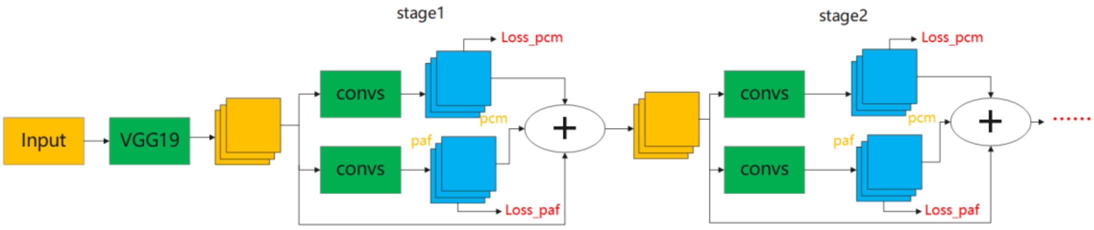

Figure 1 OpenPose original network architecture diagram

PCM is a heat map of key points, which is used to represent the location of key points. If the information of 14 key points needs to be output, then PCM will output 15 channels, and the last channel is the background channel. The purpose of this is: first, to add a supervised information, which is beneficial to the learning of the network; second, the background channel continues to be used as the input of the next stage, which is beneficial to get better semantic information in the next stage.

PAF is the core of OpenPose and is the most important feature that distinguishes OpenPose from other keypoint detection frameworks. It is used to express the affinity between different joint points, different joints belonging to the same person have high affinity, while joints between different people have low affinity. OpenPose is a bottom-up human pose estimation network, first regardless of whether the joint points belong to the the same person, first detects all the joint points, and then determines which joints have a high affinity and classifies them as the same person. Figure 2 shows the internal structure of the partial affinity prediction network.

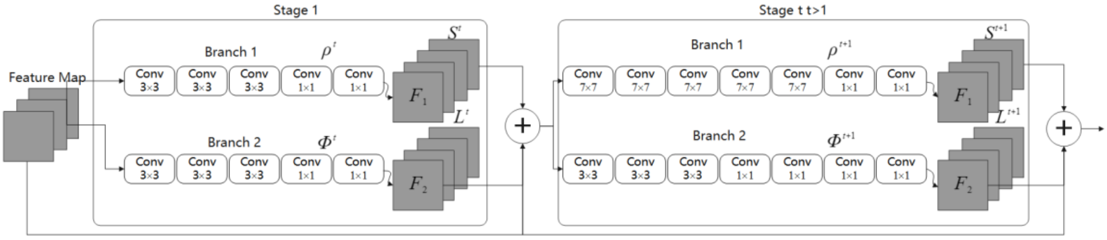

Figure 2 Internal structure of partial affinity prediction network

The first stage of the partial affinity network uses a 3 × 3 convolutional kernel, and the second stage uses a 7 × 7 convolutional kernel in order to obtain a larger perceptual field, the predicted values S^t and L^t of the two branches are connected at the end of each stage with the original feature map F as the input to the next stage, and the detailed expressions are shown in equation (1).

$$\begin{aligned} S^t &= \rho^t(F, S^{t-1}, L^{t-1}), t \geq 2 \\ L^t &= \Phi^t(F, S^{t-1}, L^{t-1}), t \geq 2 \end{aligned} \tag{1}$$

The feature extraction network of the original OpenPose model is the VGG19 convolutional neural network, and VGG19 cannot run on mobile and embedded devices due to its deep network structure and thus high memory requirements and computation. In order to apply this badminton player body pose estimation algorithm on lightweight mobile devices, therefore, in this paper, the network structure of VGG19 is changed to MobileNet network [15].

III. METHOD DESIGN

Figure 3 shows the overall framework of the pose estimation network designed in this paper. In order to embed the model in mobile devices and realize the function of real-time detection, the original backbone network VGG19 of OpenPose is changed to MobileNet network in this paper, which makes the overall network structure more lightweight; at the same time, the polarized self-attention mechanism is introduced at the front and back ends of the MobileNet backbone network, which can ensure the high resolution of the input features and the detection accuracy as much as possible with reasonable overhead.

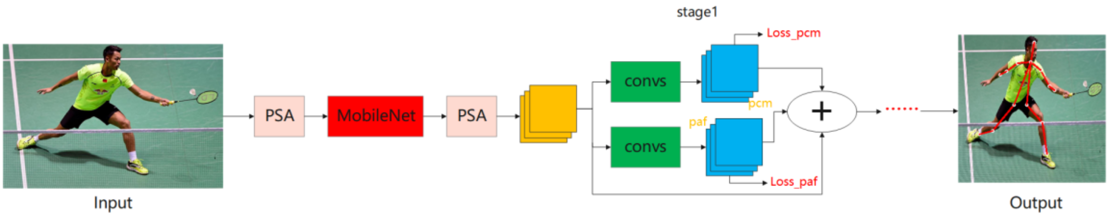

Figure 3 Lightweight human critical point detection network

3.1 Backbone Network

MobileNet network is proposed by google team in 2017, focusing on lightweight CNN network in mobile or embedded devices. Compared with the traditional convolutional neural network, the parameters and operations of the model are greatly reduced with a small decrease in accuracy, compared with VGG19 accuracy is reduced by 0.9%, but the model parameters are only 1/32 of VGG.

The network structure of MobileNet is shown in Figure 4, firstly by a standard convolution of 3×3, then followed by a stacked depth separable convolution, and some of these channel-by-channel convolutions will be downsampled by step 2, using average pooling to turn the features into 1×1, adding a fully connected layer according to the size of the prediction category, and the last one is a softmax layer. TABLE I shows the computation and parameter distribution of the MobileNet network.

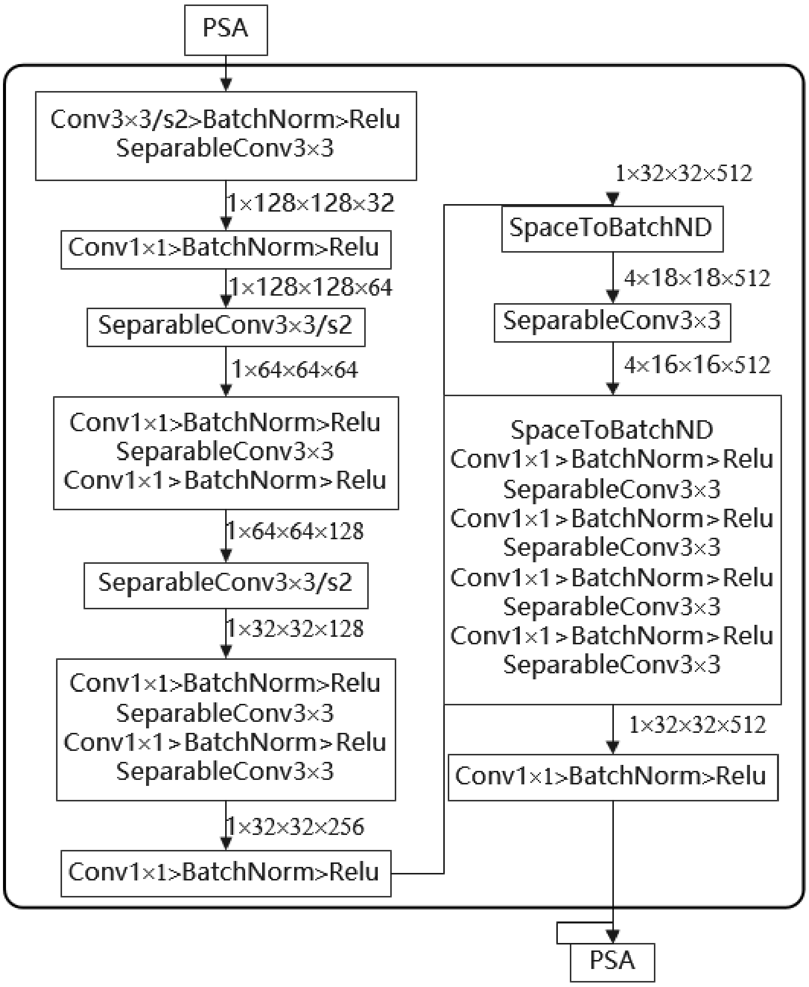

Figure 4 MobileNet network structure

TABLE I. Calculation and parameter distribution of MobileNet network

Type	Mult-Adds	Parameters
Conv 1×1	94.86%	74.59%
Conv DW 3×3	3.06%	1.06%
Conv 3×3	1.19%	0.02%
Fully Connected	0.18%	24.33%

MobileNet network has two main highlights:

The addition of Depthwise Separable Convolution, the network structure is shown in Figure 5, which is mainly divided into Depthwise Convolution (DC) and Pointwise Convolution (PC). The depthwise separable convolution can greatly reduce the parameters of the model and the amount of operations in the convolution process; two hyperparameters α and β are added, α is used to control the number of convolution kernels in the convolution layer and β is used to control the size of the input image.

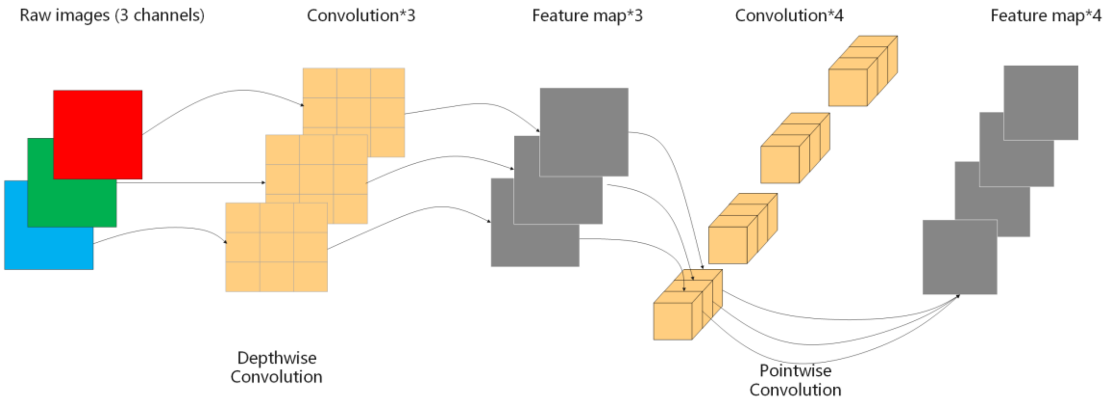

Figure 5 Structure of deep separable convolutional network

(1) Depthwise Convolution

One convolution kernel of depthwise convolution is responsible for one channel, and one channel is convolved by only one convolution kernel. This process produces exactly the same number of channels for feature extraction as the number of channels in the input. Assuming that the size of the input image is D_w×D_H×C and the size of the convolution kernel is D_c×D_c×C, the channel-by-channel convolution first undergoes the first convolution operation, which is performed entirely in the two-dimensional plane, and the number of convolution kernels is the same as the number of channels in the previous layer, and the channels and convolution kernels correspond to each other, so a C-channel image generates C feature maps after the operation. The number of parameters of the convolution part is calculated as shown in equation (2).

$$N_{depthwise} = D_c \times D_c \times C \tag{2}$$

The parameters are calculated as shown in equation (3).

$$C_{depthwise} = D_c \times D_c \times (D_w - D_c + 1) \times (D_H - D_c + 1) \times C \tag{3}$$

(2) Pointwise Convolution

The number of feature maps after the depthwise convolution is completed is the same as the number of channels in the input layer, and the number of features cannot be expanded. Moreover, this operation performs the convolution operation for each channel of the input layer independently, and does not effectively utilize the feature information of different channels at the same spatial location. Therefore, pointwise convolution is needed to combine these feature maps to generate a new feature map.

Article History: Received: 30 March 2022, Revised: 8 April 2022, Accepted: 15 April 2022, Publication: 30 April 2022

The pointwise convolution operation is very similar to the regular convolution operation, which has a convolution kernel of size 1×1×C, and C is the number of channels in the previous layer. So the convolution operation here will combine the feature maps from the previous step weighted in the depth direction to generate a new feature map. Assuming that N feature maps are output, the number of parameters involved in the convolution in this step is calculated as shown in equation (4) since 1×1 convolution is used.

$$\mathrm{N_{depthwise}} = 1 \times 1 \times \mathrm{C} \times \mathrm{N} \qquad (4)$$

The parameters are calculated as shown in equation (5).

$$\mathrm{C_{depthwise}} = \mathrm{D_c} \times \mathrm{D_c} \times \mathrm{D_W} \times \mathrm{D_H} \times \mathrm{C} \times \mathrm{N} \qquad (5)$$

After point-by-point convolution, the same N feature maps are output, with the same dimensionality as the output of regular convolution.

3.2 Polarized Self-Attention Module

In the critical point detection task, an upsampling-downsampling network structure is required to change the number of channels of the feature map while deepening the network depth, but such a structure, in the process of downsampling, may lead to the loss of some of the input information. In order to be able to ensure the high resolution of the input features as much as possible with a reasonable overhead, this paper introduces the Polarized Self- Attention (PSA) [16]. PSA not only maintains high resolution within its module, but its design of mixing softmax and sigmoid can fit a more realistic output.

PSA is plug-and-play, lightweight, simple, and efficient, and its core consists of two parts: Channel-only Self-Attention and Spatial-only Self-Attention. This paper uses a tandem approach to fuse channel branches and spatial branches, and Figure 6 shows the implementation flow of the PSA module used in this paper.

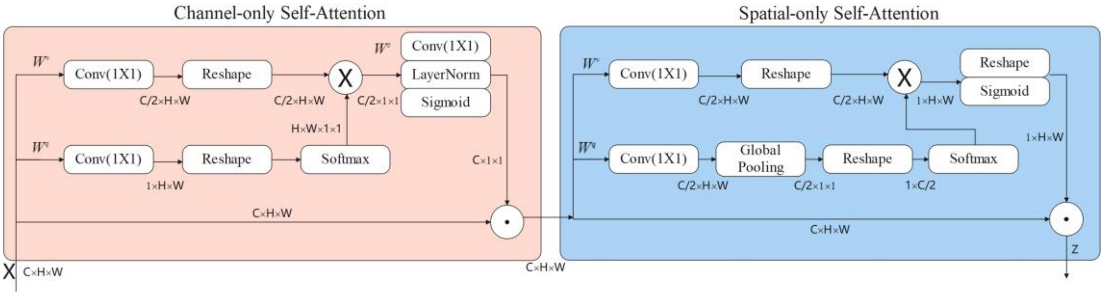

Figure 6: PSA module implementation flow

The main operation of Channel-only Self-Attention can be divided into 3 steps:

Converting the input features X into Q and V by 1 × 1 convolution, where the number of channels of feature Q is compressed to 1 and the number of channels of feature V is maintained at C/2;

Information enhancement of the feature layer Q using softmax and feature fusion of the enhanced features Q with the features V in a matrix multiplication;

The fused features are passed through a 1×1 convolutional layer and LN layer to expand the number of channels so that it is consistent with the number of channels of the input feature X. Finally, the weights of the channel branches are obtained by sigmoid nonlinear transformation, and the input of the spatial branch is calculated by weighting the input feature X.

The steps of Spatial-only Self-Attention are similar to Channel-only Self-Attention, with two main differences: for feature Q, GlobalPooling is used to compress the channel dimension; the fused features are no longer passed through 1×1 convolutional and LN layers, but are directly reshaped for normalization.

IV. EXPERIMENTAL RESULTS AND ANALYSIS

4.1 Experimental Data and Experimental Design

Therefore, in this paper, we use the "human skeletal keypoints" dataset from the AI Challenger 2017 Global Challenge [17] to train and validate the model. Each image in this dataset is labeled with 14 human skeletal keypoints, and Figure 7 shows the specific locations of these 14 human keypoints. In order to evaluate the model's pose capture of athletes in badminton scenes, 946 images in badminton scenes are screened as the dataset in this paper, and the dataset is expanded to 2000 images by data enhancement strategy, and the training set and validator are divided in the ratio of 7:3. The screened part of the test set is shown in Figure 8.

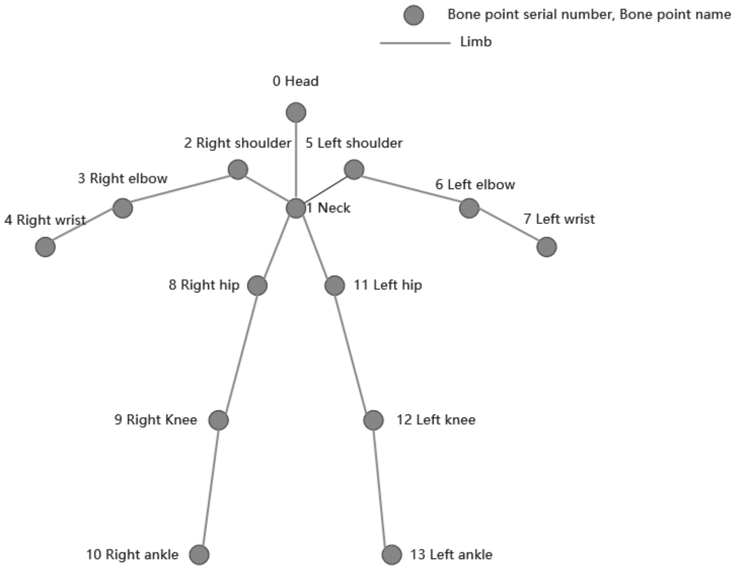

Figure 7 The specific location of the 14 key points

Figure 8 Images of some of the badminton scenes in the AI Challenger 2017 dataset

In terms of model training, this paper uses Python 3.8 as the training environment for the model, and builds a tensorflow deep learning framework under Ubuntu environment. Finally, the experiments use an NVDIA 3070 GPU to process the data and train and validate the model proposed in this paper.

4.2 Experimental Results and Analysis

To evaluate the accuracy of the lightweight OpenPose model proposed in this paper, we used relevant data from the AI Challenger 2017 Keypoint Challenge for validation. all comparison experiments were performed using the tensorflow deep learning framework and in Ubuntu and Python 3.8. In this paper, the initial learning rate is set to 5e-3 and the training iterations are 200 epochs with a batch size of 16. mean average precision (mAP) and Frames Per Second (FPS) are chosen as evaluation metrics for the skeletal point detection network, and Equation 6 represents the calculation of these metrics Equation 6 shows the calculation process of these metrics.

$$P=\frac{TP}{TP+FP},R=\frac{TP}{TP+FN},mAP=\int_0^1 P(R)dR,FPS=\frac{numbers}{seconds} \tag{6}$$

Where P denotes accuracy; R denotes recall; TP, FP, and FN denote true positives, false positives, and false negatives, respectively; numbers and seconds denote the number of images detected and the time spent on detection, respectively.

The experimental comparison results of the improved model with the original OpenPose model are shown in TABLE II, where mAP0.5 and mAP0.75 denote the mean average accuracy with threshold values of 0.5 and 0.75, respectively. Although there is a slight decrease in the average accuracy, within 3 percentage points, the improvement in detection speed is obvious, which fully demonstrates the advantages of the deep separable convolution. Taking N DK-sized convolution kernels as an example, the number of parameters of the depth-separable convolution is only $\frac{1}{N}+\frac{1}{D_K^2}$ of the standard convolution with the number of input channels M, which greatly shortens the prediction time of the model and improves the advantage of the model in capturing athletes' pose movements in real time.

TABLE II. Model comparison results

Model	*mAP0.5 (%)*	*mAP0.75 (%)*	*FPS*
OpenPose	86.23	73.13	14
Ours	85.79	70.57	23

Figure 9 shows some of the test results of our algorithm on the AI Challenger dataset. Although our model is slightly inferior to the original OpenPose model in terms of average accuracy, it can still detect the key points of the human body well and capture the pose of athletes in various states, so that our model can ensure good recognition accuracy for skeletal point detection and also meet the requirement of capturing the pose of athletes in real time when they are moving fast in training and competition scenarios.

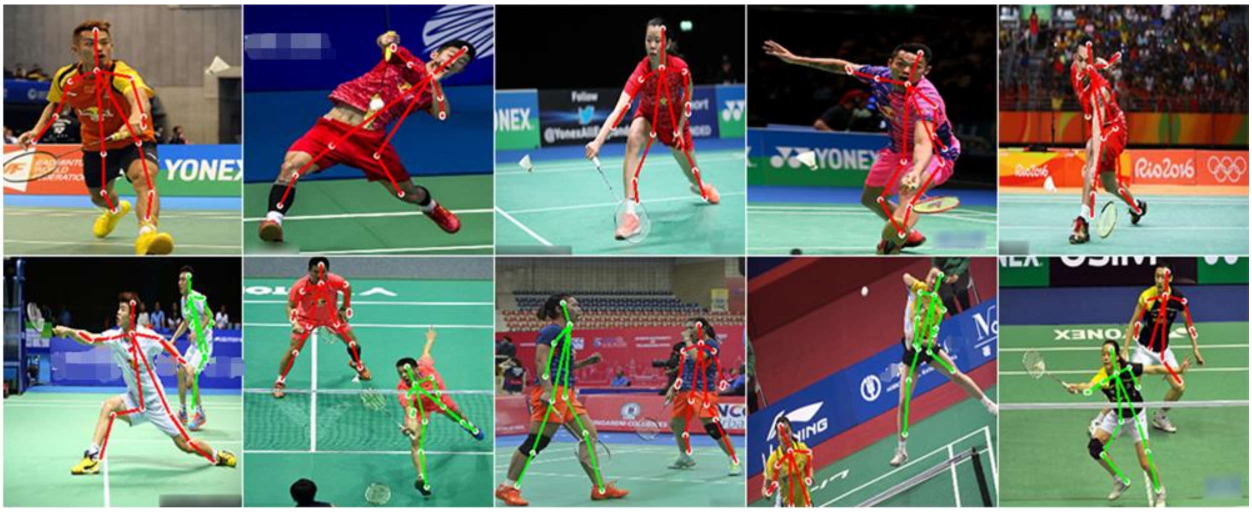

Figure 9: Partial detection results of the model after network optimization

V. CONCLUSION

In order to estimate the pose of fast-moving badminton players in real time, it is necessary to apply the human pose estimation algorithm to embedded devices. In this paper, the original OpenPose model is optimized to achieve a lightweight model by replacing its original backbone network VGG19 with MobileNet, a lightweight network proposed by Google team in 2017, and adding a Polarized Self-Attention (PSA) mechanism to both the front and back ends of the MobileNet network to ensure that the internal module remains high resolution. Attention (PSA) to ensure that the module maintains high resolution internally, and its design of mixing softmax and sigmoid can fit a more realistic output. The final experiments show that the optimized model greatly improves the estimation speed with a small reduction in estimation accuracy, and can be applied in embedded devices to provide a comprehensive evaluation of athletes' performance to guide them to adjust their sports posture in a more scientific and reasonable way.

The shortcoming of this study is that the stance of badminton players' hands and legs are not separately identified, and the power of wrist joints and the correctness of the pace play a crucial role in the improvement of badminton players' performance. Therefore, in addition to the estimation of the whole human posture, it is particularly important to perform stepwise estimation of hand and leg postures alone, and a deeper study of lightweight skeletal point detection networks with higher accuracy to estimate the key points of the hands and legs is the focus of subsequent research.

REFERENCES

[1]Lu Hongtao, Zhang Qinchuan. Applications of Deep Convolutional Neural Network in Computer Vision. Journal of Data Acquisition and Processing, 2016, 31(01): 1004-9037.

[2]LU jian, YANG Tengfei, ZHAO Bo, et al. A Review of Deep Learning-Based Human Pose Estimation. Laser & Optoelectronics Progress, 2021, 58(24): 69-88.

[3]Zhou Yan, LIU Ziqin, ZENG Fanzhi, et al. Survey on Two-Dimensional Human Pose Estimation of Deep Learning. Journal of Frontiers of Computer Science and Technology, 2021, 15(04): 641-657.

[4]Zong Li-bo, Song Yi-fan, Wang Yi-ming, et al. Human Pose Estimation in Sports Video Analysis: a Survey. Journal of Chinese Computer Systems, 2021, 41(08): 1751-1757.

[5]Wang Yuan. Human motion posture attitude estimation and recognition based on deep natural network. chengdu: School Mechanical and Electrical Engineering, 2020.

[6]CHEN Yao-Dong, LI Ren-Fa, LI Shi-Ying, et al. A Combined Grammar for Object Detection and Pose Estimation. CHINESE JOURNAL OF COMPUTERS, 2014, 37(10): 2206-2217.

[7]FELZENSZWALB P. Pictorial structure for object recognition. International Journal on Computer Vision, 2005, 61(1):55-79.

[8]FANGHS, XIE SQ, TAI YW, et al. RMPE: regional multiperson pose estimation // Proceedings of the 2017 IEEE International Conference on Computer Vision. Piscataway: IEEE,

[9]CAO Z, SIMON T, WEI S E, et al. Realtime multi-person 2D pose estimation using part affinity fields//Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition.Piscataway:IEEE,2017:1302-1310.

[10]YANG Hong-hong, Wang Liu-li, ZHANG Yu-mei, et al. Hierarchical Dance Pose Estimation Algorithm Based on Sequential Multi-Scale Feature Fusion. Acta Electronica Sinica, 2021, 49(12): 2428-2436.

[11]ZHANG Yu, DIAO Yanan, LIANG Shengyun, et al. Cognitive-motion Rehabilitation Medical Robot Application Design. Information and Control, 2021, 50(06): 740-747.

[12]Li Yifan, Yuan Longjian, Wang Rui, et al. Improved lightweight human action recognition model based on OpenPose. ELECTRONIC MEASUREMENT TECHNOLOGY, 2022, 45(01): 89-95.

[13]Liu Yucong, Xu Shuocheng, Song Shuaichao, et al. Traditional Wushu Action Recognition and Comparison Based on OpenPose. Electronic Component and Information Technology, 2021, 5(3): 126-128.

[14]MENG Linshen, Li Jianying, HAO Huidong. Biomechanics Analysis of Elite Badminton Players’ Front Step Technique. JOURNAl OF XI AN PHYSICAL EDUCATION UNIVERSITY, 2018, 35(06): 722-730.

[15]Andrew G Howard, Menglong ZHU, Bo CHEN. Dmitry Kalenichenko, et al. Mobilenets: Efficient convolutional neural networks for mobile vision applications. arXiv:1704.04861, 2017.

[16]Liu H, Liu F, Fan X, et al. Polarized self-attention: Towards high-quality pixel-wise regression. arXiv preprint arXiv:2107.00782, 2021.

[17]Wu J, Zheng H, Zhao B, et al. Ai challenger: A large-scale dataset for going deeper in image understanding. arXiv preprint arXiv:1711.06475, 2017.