

Research Article

Analysis of Physical Expansion Training Based on Edge Computing and Artificial Intelligence

Zhongle Liu 

Jilin Agricultural Science and Technology University, Jilin 132101, China

Correspondence should be addressed to Zhongle Liu; liuzhongle@jlnku.edu.cn

Received 27 April 2021; Revised 20 May 2021; Accepted 25 May 2021; Published 2 June 2021

Academic Editor: Jianhui Lv

Copyright © 2021 Zhongle Liu. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The effective development of physical expansion training benefits from the rapid development of computer technology, especially the integration of Edge Computing (EC) and Artificial Intelligence (AI) technology. Physical expansion training is mainly based on the collective form, and how to improve the quality of training to achieve results has become the content of everyone's attention. As a representative technology in the field of AI, deep learning and EC evolving from traditional cloud computing technology are all well applied to physical expansion training. Traditional EC methods have problems such as high computing cost and long computing time. In this paper, deep learning technology is introduced to optimize EC methods. The EC cycle is set through the Internet of Things (IoT) topology to obtain the data upload speed. The CNN (Convolutional Neural Network) model introduces deep reinforcement learning technology, implements convolution calculations, and completes the resource allocation of EC for each trainer's wearable sensor device, which realizes the optimization of EC based on deep reinforcement learning. The experiment results show that the proposed method can effectively control the server's occupancy time, the energy cost of the edge server, and the computing cost. The proposed method in this paper can also improve the resource allocation ability of EC, ensure the uniform speed of the computing process, and improve the efficiency of EC.

1. Introduction

Physical training generally refers to all physical activities that maintain and develop proper physical expansion and improve physical health through exercise. Regular physical training can activate the body's immune system and prevent or improve some civilization diseases, such as cardiovascular disease, type 2 diabetes, and obesity. It can also improve mental health, reduce depression, increase resistance ability to stress, improve sleep quality, improve insomnia problems, and help form positive self-esteem. Regular exercise is one of the keys to maintaining health, and it has a significant contribution to maintaining a healthy weight, digestive system, bone density, muscle capacity, free movement of joints, physiological function, reducing the chance of facing surgery in the future, and strengthening the immune system.

Physical fitness is the human body's ability to overcome resistance, rapid movement ability, continuous work (exercise) ability, coordinated movement ability, and sensitive

and accurate movement ability that the human body shows in exercise, labor, and life [1]. It can be considered that physical fitness not only reflects the basic functional capabilities of human activities but also reflects the basic functional capabilities of human labor and life. Physical training is an indispensable basic athletic ability for work and life. It is conducive to mastering complex technical movements and improving exercise effectiveness, withstand heavy load training and high-intensity exercise, and maintain a stable and good mentality in daily training and competition status. The physical fitness training is a team style physical training with physical fitness as the guide, games as a tool, and mental fitness as the main purpose. Physical expansion training is rarely completed by individuals, usually in a collective form. The relationship within the group is directly related to the actual training benefits. In addition to the conventional training methods, the content of physical development training should also include some other contents, such as handstands and walking backward.

Physical expansion training is mainly based on the collective form, and how to improve the quality of training has become the content of everyone's attention. In recent years, with the rapid development of social economy and science and technology on a global scale, many emerging technologies have continuously emerged in information and communication technology industry [2, 3]. Among them, two representative technologies are widely regarded as having a huge impetus and far-reaching influence on the human economy and society. First, as a representative technology in the field of AI, deep learning has benefited from advances in algorithms and data sets [4]. It has been developed by leaps and bounds in recent years and is used in unmanned driving, e-commerce, smart home, and smart finance. The field has played a big role, profoundly changed people's lifestyle, and improved production efficiency. The other technology is EC evolved from traditional cloud computing technology. Compared with cloud computing, EC sinks strong computing resources and efficient services to the edge of the network, thereby having lower latency, lower bandwidth usage, higher energy efficiency, and better privacy protection. The introduction of EC and AI into physical expansion training can better help people train [5].

The rapid development of the IoT has brought us into the postcloud era, which will generate a lot of data in our daily lives [6, 7]. IoT applications may require extremely fast response time, data privacy, and so on. If the data generated by the IoT is transmitted to the cloud computing center, the network load will be increased, the network may cause congestion, and there will be a certain data processing delay. With the increase of mobile devices and the increase of camera deployment in cities, the use of video to achieve a certain purpose has become a suitable means, but the cloud computing model is no longer suitable for this kind of video processing, because a large amount of data in the network transmission in the video may cause network congestion, and the privacy of video data is difficult to guarantee. Therefore, EC is proposed to allow the cloud center to delegate related requests. Each edge node processes the request combined with local video data and then only returns the relevant results to the cloud center. This not only reduces network traffic but also guarantees privacy to users to some extent [8]. EC refers to processing and analyzing data at the edge of the network, which can reduce request response time, improve battery life, reduce network bandwidth, and ensure data security and privacy. An edge node is any node with computing resources and network resources between the source of data generation and the cloud center. For example, smart wearable devices are the edge nodes between people and cloud centers. In an ideal environment, EC refers to analyzing and processing data near the source of data generation, without data circulation, thereby reducing network traffic and response time. In order to quickly update the training model and improve efficiency.

Physical expansion training is completed through groups; everyone in the group is equipped with a wearable sensor. The wearable sensor can analyze the exercise quality of each trainer. Two epoch-making new technologies, AI and EC, are currently facing bottlenecks in their further

development. On the one hand, for deep learning technology, because it requires high-density calculations, current intelligent algorithms based on deep learning usually run in cloud computing data centers with powerful computing capabilities. With the high popularity of mobile terminal devices consideration, how to effectively deploy deep learning models in resource-constrained terminal devices has attracted great attention from academia [9, 10]. It has aroused great attention from academia and industry [9, 10]. On the other hand, with the sinking and decentralization of computing resources and services, EC nodes will be widely deployed at network edge access points (such as cellular base stations, gateways, and wireless access points). The high-density deployment of EC nodes also brings new challenges to the deployment of computing services; users usually have mobility; therefore, when the user moves between different nodes coverage frequently, whether computing services should be with the trajectory of the mobile user migration, this is a dilemma problem, because although service migration can reduce delay and improve with experience, it will bring additional costs such as bandwidth and energy consumption.

The development bottlenecks faced by AI and EC can be alleviated through synergy. On the one hand, for deep learning, mobile devices running deep learning applications offload part of model inference tasks to adjacent EC nodes for calculations, thereby cooperating with terminal devices and edge servers to integrate the local computing capabilities and strong computing capabilities of the two complementary advantages. In this way, because a large number of calculations are executed on EC nodes with strong computing power adjacent to the mobile device, the resource and energy consumption of the mobile device itself and the delay of task inference can be significantly reduced, thereby ensuring good user experience. On the other hand, in view of the dynamic migration and placement of EC services, AI technology is also promising. Specifically, based on high-dimensional historical data, AI technology can automatically extract the mapping relationship between the optimal migration decision and high-dimensional input, so that when a new user location is given, the corresponding machine learning model can quickly map it to the optimal migration decision. In addition, based on the user's historical trajectory data, AI technology can also efficiently predict the user's movement trajectory in the short term in the future, thereby realizing predictive edge service migration decisions and further improving the service performance of the system. In general, EC and AI will generate a new paradigm of "edge intelligence," which will generate a large number of innovative research opportunities.

Starting from the dimension of EC combined with AI, the main contribution of the paper is to introduce deep reinforcement learning technology to EC and propose a method for EC to drive real-time deep reinforcement learning. The rest of the paper is organized as follows. Section 2 analyzes and summarizes domestic and foreign research work in physical expansion training using EC and AI. Section 3 proposes that EC drives real-time deep

reinforcement learning methods. The experimental results are reported in Section 4, and finally, Section 5 concludes this paper.

2. Related Work

AI has improved people's quality of life and living standards. At the same time, with the rapid development of the mobile Internet and the IoT industry, as a novel combination of AI application and EC, intelligent edge system is promising by researchers in the field of AI and network computing. The research on intelligent edge system based on EC architecture is becoming more and more important.

Traditional EC research mostly considered the problem of task offloading, that is, whether the tasks generated by the end device should be handed over to the network edge device for calculation and processing. There has been a lot of research in this area. In the research direction of task offloading, a lot of work focuses on the energy consumption optimization problem of mobile devices [11]. In [12, 13], considering the task offloading problem in energy harvesting systems, the study in [12] proposed an effective based on the resource management algorithm of reinforcement learning; the algorithm obtains the optimal strategy of dynamic offloading through online learning. The study in [13] proposed a low-complexity online algorithm based on the Lyapunov optimized dynamic computing offloading algorithm, which can make decisions for task offloading only by relying on the current system state. In [14, 15], the authors studied the service caching mechanism in EC systems, and some fault-tolerant mechanisms for EC systems were also being studied [16–18]. The study in [19] proposed a series of methodologies, using association rule mining technology to analyze the physical fitness index of basketball players, using data processing and database management functions, which can also solve the management of athletes' physical ability indicators and assist coaches in managing players and calculating training results to improve the efficiency of data processing. According to the data mining technology analysis of the player's physical training data, from the aspect of competitive sports, the goal of training was to create excellent sports performance, as well as the most basic competitiveness and the most controllable factors for the improvement of the player's physical ability. Physical training was the basic way for coaches to know the physical fitness of players. The coaches regularly test the physical performance of the players. According to different test calculation standards, they calculate the results of the physical fitness test for each item of each player. Later, based on their own experience, they would evaluate the physical fitness of the players to formulate a reasonable effective training plan to train. But with the accumulation of test data, it would become more and more difficult to analyze this pile of data with manual management work, and the commonly used computer data processing and database management functions can solve the management of the player's physical test data. It cannot find potential knowledge outside the database, and it cannot provide effective evaluation and prespeculation of the player's physical condition.

In terms of the design of an intelligent physical training platform, there was a ZigBee-based physical training platform designed by Huo et al. of Shenyang Ligong University. The platform achieved the functions of data collection and transmission at the same time [20]. With the rapid popularity of mobile devices, many wearable sports data collection products for ordinary sports enthusiasts have appeared on the market. For example, Huawei smart bracelets, Nike+, and Adidas miCoach are powerful, but they are only for professionals. The physical team or a certain kind of training program, and these products required expensive auxiliary equipment, which was not suitable for physical expansion training. In terms of training method recognition, the study in [21] proposed a hidden condition random field object recognition model based on the maximum boundary value and combined a large number of global and local features to distinguish different actions. In the training process, people mainly pay attention to the impact of training on physical functions. Different amounts of training have different effects on physical functions [22]. When the energy consumption is small, the body's metabolism is higher; when the energy consumption is large or even too large, the human body's metabolism is large. Although it has reached the training volume, it will also cause excessive energy consumption and produce some harmful effects. Metabolic wastes, in severe cases, can even cause shock or crushing death, which has harmful effects on body functions. It can be seen that only when the energy consumption during physical training is controlled within a reasonable range, can it have a positive impact on the changes in human body function.

3. EC Drives Real-Time Deep Reinforcement Learning Method

In the physical development training, each trainer is equipped with wearable sensors. These devices generate a large amount of data, which makes the traditional computing framework and the amount of data incompatible. At the same time, the trainer cannot get real-time feedback on the network transmission speed delay. Only real-time feedback can help the development of physical expansion training, so as to establish an effective training model for each trainer and adjust the training method in time. The delay makes it impossible to realize the cloud transmission of IoT data. A large amount of data is directly consumed at the edge of the network. Therefore, it is necessary to carry out calculations on the edge of the IoT. In physical development training, each trainer is an edge node.

As one of the mainstream technologies in the field of AI, deep learning has been strongly sought after by academia and industry in recent years [23, 24]. Since deep learning models require a lot of calculations, intelligent algorithms based on deep learning usually exist in cloud computing data centers with powerful computing capabilities. With the rapid development and popularization of mobile terminals and IoT devices, how to break through the resource limitations of terminal devices so as to efficiently run deep learning models on resource-constrained terminal devices has attracted a lot of attention. To solve this problem,

consider the idea of EC to empower AI and use the characteristics of near real-time computing of EC to reduce the delay and energy consumption of deep learning model inference. For this reason, in this research, deep reinforcement learning is integrated into EC to reduce computing costs. Deep reinforcement learning combines the perception ability in deep learning with the decision-making ability in reinforcement learning and optimizes the original EC through AI to ensure the validity and feasibility of the optimized results.

EC is a technology which deploys computing tasks between the cloud and the terminal. The characteristics of EC and its proximity to the device are destined to have the advantages of real-time processing, so it can better support the real-time processing and execution of local services. EC directly filters and analyzes the data of the terminal equipment, which saves energy and saves time and efficiency. In other words, some terminals will offload the computing task to the EC device and perform the computing task through the resources allocated by the edge device. In the cloud computing model, terminal equipment is the data-consuming role. Data producers (such as YouTube) publish data to the cloud, and data consumers (such as mobile phones) request data to obtain cloud data. This is a traditional cloud computing model, but with the popularity of IoT devices, the physical expansion training in this paper, terminal trainers use wearable sensors to generate a large amount of data, and the processing and transmission of these data will encounter some problems. EC is committed to solving these problems, so EC is integrated with the IoT technology, the amount of data is very large, it needs to take up a large upload bandwidth, so the data needs to be processed on the device side and processed into a suitable format for transmission, and the computing task is processed on the terminal device. Therefore, the terminal is no longer pure data consumption but plays a role in data generation. In summary, EC uses the processing power of the LAN gateway to process more real-time information.

3.1. EC Cycle Setting. The EC cycle needs to analyze the topology of the IoT and set the EC cycle of the IoT based on the analysis results. Under normal circumstances, the topology of the IoT can be divided into two parts: a flat network structure and a hierarchical structure. Aiming at the relevant characteristics of the IoT, in this paper, the hierarchical results are used as the research object.

According to the trainer's wearable device business data and the randomness of renewable energy, the continuous time scale is set to divide the time at equal intervals, and the time interval is set to Δt , which is the calculation decision period at equal intervals. Using the above period, the decision time and the decision period Δt can be dynamically adjusted to meet the complexity and variability of EC in the IoT. At each calculation decision point, the generation rate of business data is set to $bg_0(t)$ and $\eta(t)$; then, the accumulated data size and energy value of the IoT in the calculation period of equal time interval can be expressed as

$$bg(t) = bg_0(t)\Delta t, \quad (1)$$

$$E(t) = x(t)\Delta t. \quad (2)$$

Among them, $bg(t)$ is defined as the amount of data accumulated by the trainer's wearable device during the calculation period, and $E(t)$ represents the accumulated energy value of the trainer's wearable device during the calculation period. Through the above formula, the data generated in the edge calculation cycle is controlled. In the case of different calculation cycles, in order to facilitate the development of calculations, the generation rate of the set business data and the energy attainment rate are both reflected in the form of independent and identical distribution. Set the bandwidth of the wireless broadband in the IoT of the trainer's wearable device as B , and there is only one base station in an IoT; ignoring the interference of the base station, the data upload rate U_i of the IoT can be expressed as

$$U_i = \frac{Bp_i^l}{E(t)} + \frac{bg(t)f_i^l}{\chi^2} \log \text{ber}_{\text{up}}. \quad (3)$$

Among them, p_i^l represents the computing power of the wearable device of each trainer, f_i^l represents the local computing power of the wearable device, χ^2 is the variance of Gaussian white noise, and ber_{up} is the target bit error rate. Using the above parts, the design part of the EC cycle of the IoT is completed, and this result is used as the data basis for constructing the EC execution process.

3.2. EC Execution Process. Using the above analysis results, as the basis for the construction of the EC execution process of each trainer's wearable device, deep reinforcement learning technology is used to complete the EC execution process.

For the local computing part $(1 - x_i)q_i$ of each trainer's wearable device, define t_i as the local execution delay; this part contains the processing time of the server; set cf_i^t as the CPU frequency for the calculation; then, the execution delay de_i^t can be expressed as

$$de_i^t = \frac{(1 - x_i)q_i l_i}{cf_i^t} + \log \left(1 + \frac{|cg_i|}{\Gamma(\text{ber}_{\text{up}})} \right). \quad (4)$$

Among them, l_i is the length of the communication channel. $\Gamma(\text{ber}_{\text{up}})$ represents the margin of signal-to-noise ratio introduced to meet the uplink target error rate. Set EC_i^t as the energy consumption during local execution. According to formula (4), it can be expressed as

$$EC_i^t = \rho_i cf_i^t (1 - x_i) q_i l_i N_0. \quad (5)$$

Among them, ρ_i TT is set to the energy density in the IoT, which represents the energy consumed in the decision-making cycle during the calculation process, and N_0 represents the noise power of the channel. Considering that the change of cf_i^t will affect the change of calculation energy consumption, the dynamic voltage scaling technology is used to set the overall calculation time ω_i^t , and the local

calculation frequency cf_i^t is reasonably allocated. The sub-part can be expressed as

$$cf_i^t = \min \left\{ \frac{(1-x_i)q_i^L i}{\omega_i^t}, cf_i^{t_{\max}} \right\}. \quad (6)$$

Use the above formula to control the execution process of EC. Ensure that the energy consumption of EC matches the characteristics of each trainer's wearable device. According to the data transmission speed calculated in formula (3), the delay time of the calculation process can be calculated as follows:

$$t_i^0 = \frac{q_i}{\omega_i}. \quad (7)$$

Using the above formula to control the time during the execution of EC, according to the EC execution process designed in this part, in-depth reinforcement learning technology is integrated into the EC resource allocation of each trainer's wearable device.

3.3. EC Resource Allocation. In this paper, the CNN model in deep reinforcement learning technology is used as the design basis to realize the EC resource allocation of the IoT, and the convolution processing is mainly used to complete the rational allocation of resources [25, 26].

According to the relevant knowledge of the signal and the network, the convolution operation of the two signals in the decision time period can be embodied in the form of a formula as follows:

$$\text{conv}(t) = s(t) * w(t). \quad (8)$$

Among them, $s(t)$ and $w(t)$, respectively, represent the signals in edge calculation. The discrete sequences $f(n)$ and $g(n)$ in resource allocation are obtained through the translation, multiplication, and integration results of these two signals in the time period, and the convolution results are embodied in discrete forms.

$$\text{conv}(n) = f(n) * g(n) = \sum_{m=0}^{N-1} f(n)g(n-m). \quad (9)$$

Using the above formula, set the number of connections in the EC of each trainer's wearable device. According to the parameter characteristics of the CNN model, set the weight and bias in the allocation process as the number of connections; then, the number of parameters in the edge calculation can be expressed as $ct * wi_k * he_k + 1$, where ct is the number of channels in the calculation and wi_k and he_k are the width and height of the convolution kernel. Through this formula, the calculation amount of the calculation process can be obtained as $ct * wi_k * he_k$, and the rationalized distribution of the calculation amount can be expressed as

$$\rho_i(t) = \frac{ct * wi_k * he_k}{\text{conv}(n)}. \quad (10)$$

Using the above formula, complete EC resource allocation. Connect this part with the design part above in an orderly

manner to realize the application of deep reinforcement learning in EC of the IoT. So far, the design of real-time deep reinforcement learning method driven by EC is completed.

4. Experiment Design and Result Analysis

The experiment evaluates the application effect of the previously designed deep reinforcement learning in EC through real data sets. In addition, this paper also compares the proposed method with three methods in server occupancy time, server power consumption, and calculate waiting time.

4.1. Data Set Design. Python 3.8 is used to implement the calculation process of this experiment. In order to make the experimental results more convincing, Google Cluster is used as the data set, and the task samples are constructed using attributes such as CPU request and memory request. In the simulation scenario of this paper, suppose there are 10 edge nodes, that is, 10 trainers, and each trainer is equipped with wearable sensors. The bandwidth, computing power, and computing power per unit time of the edge node (each trainer) can be seen from Table 1 for consumption. At the same time, suppose that the generation rate of business data is 0.36 and 0.73 for $bg_0(t)$ and $\eta(t)$, respectively, the CPU frequency cf_i^t is 4 GHz, $\Gamma(\text{ber}_{\text{up}}) \leq 6$ dB, the local computing power f_i^t of the wearable sensor device equipped by the trainer is 30 Mb/s, and the bit error rate ber_{up} is 0.48.

The above-mentioned setting part is used as the preparation stage of the experiment, using the above-mentioned setting results to complete this experiment and comparing the difference between the calculation method after using the deepening intensity learning and the method before using this technology.

4.2. Contrast Indicators. The content of the experiment is set as the performance comparison of the calculation method, and the calculation cost is the focus of the experiment. The so-called computing cost is composed of edge server occupancy time, edge server energy cost, and average computing waiting time. In the experiment, in order to increase the effectiveness, the computing environment is set to both the full work of the edge server and the part of the edge server to verify the applicability of each calculation method. In the course of the experiment, the calculation performance of each algorithm is studied in the form of a uniform increase in the number of tasks, and the specific results are embodied in the form of images.

4.3. Analysis of Experimental Results

4.3.1. Server Occupancy Time. In order to verify the server occupancy time of different methods in EC, compare the server occupancy time after the optimization method of deep reinforcement learning, the improved cat group algorithm, and the edge-cloud collaborative IoT optimization method, and the results are shown in Figure 1.

According to Figure 1, it can be seen that the edge server takes different time under different methods. In Figure 1(a),

TABLE 1: Bandwidth, computing power, and computing energy consumption of edge nodes.

No.	Bandwidth (MHz)	Calculated ability (Mb/s)	Calculated energy consumption per unit time (J)
1	100	150	0.002
2	100	200	0.003
3	100	100	0.001
4	150	150	0.002
5	150	200	0.003
6	150	100	0.001
7	150	150	0.002
8	200	150	0.002
9	200	200	0.003
10	200	100	0.001

when the number of tasks is 20, the occupancy time of the edge server without optimization is 44.8 s, and the occupancy time of the edge server of the edge-cloud cooperative optimization method is 26.7 s. Improved cat group algorithm edge server takes 23.5 s, and the server occupancy time of the deep reinforcement learning optimization method is 7.5 s. Although the three methods can effectively reduce the server occupancy time, the occupancy time of the method in this paper is significantly lower than that of other methods, which shows that the trainer is equipped with wearable sensor equipment to occupy less server time.

In Figure 1(b), in some working environments, when the number of tasks is 30, the edge server that is not optimized takes 38.6 s, and the edge server of the edge-cloud cooperative optimization method takes 24.7 s. Improved cat group algorithm edge server takes 18.2 s, and the server occupancy time of the deep reinforcement learning optimization method is 2.7 s. When the number of tasks is 70, the unoptimized edge server takes 49.8 s, the edge-cloud cooperative optimization method takes 42.5 s, and the improved cat group algorithm IoT optimization method takes 33.5 s. The server occupancy time of the deep reinforcement learning optimization method is 3 s. In some working environments, the server occupancy time of the method in this paper is also significantly lower than other methods, which shows that the method in this paper has higher computational efficiency and strong applicability.

With the continuous increase of the number of tasks, in two different edge server working states, the algorithm using deep reinforcement learning technology can ensure the normal operation of the server. The use of deep reinforcement learning technology can effectively control the server's occupancy time so that each trainer's wearable device has more time to process local business and establish an optimized training model to better achieve the purpose of physical expansion.

4.3.2. Server Power Consumption. Based on the above, the server power consumption of the above three methods is

obtained through statistics, and the results are shown in Table 2.

Analysis of Table 2 shows that the server energy costs vary from different methods. The energy cost of the method in this paper is significantly lower than the other two methods. The use of deep reinforcement learning technology in the calculation method can effectively control the energy cost of the edge server in the calculation process, so as to ensure the calculation cost and minimize the energy consumption of the wearable sensor device.

4.3.3. Calculation Waiting Time. In order to further verify the calculation efficiency of different methods, the average calculation waiting time experiment under different tasks is added, and the results are shown in Figure 2.

Analyzing Figure 2 shows that under different number of tasks, the calculation waiting time is different. When the number of tasks is 5, the average calculation waiting time of the deep reinforcement learning method is 1 ms, the average calculation waiting time of the improved cat group algorithm is 3 ms, and the average calculation waiting time of the edge-cloud cooperation method is 3.2 ms. When the number of tasks is 40, the average calculation waiting time of the deep reinforcement learning method is 1.25 ms, the average calculation waiting time of the improved cat group algorithm is 4 ms, and the average calculation waiting time of the edge-cloud cooperation method is 3.9 ms. The method in this paper can improve the resource allocation ability of EC, ensure the uniform speed of the computing process, and improve the efficiency of EC.

Integrating the results of the average calculation waiting time, the energy cost results of the edge server and the edge server occupancy time results show that the EC method based on deep reinforcement learning designed in the paper can effectively control the computing cost and complete efficient EC during performance. In the process, the model for the trainer can be applied to the training faster, so that the physical expansion training can be carried out more effectively.

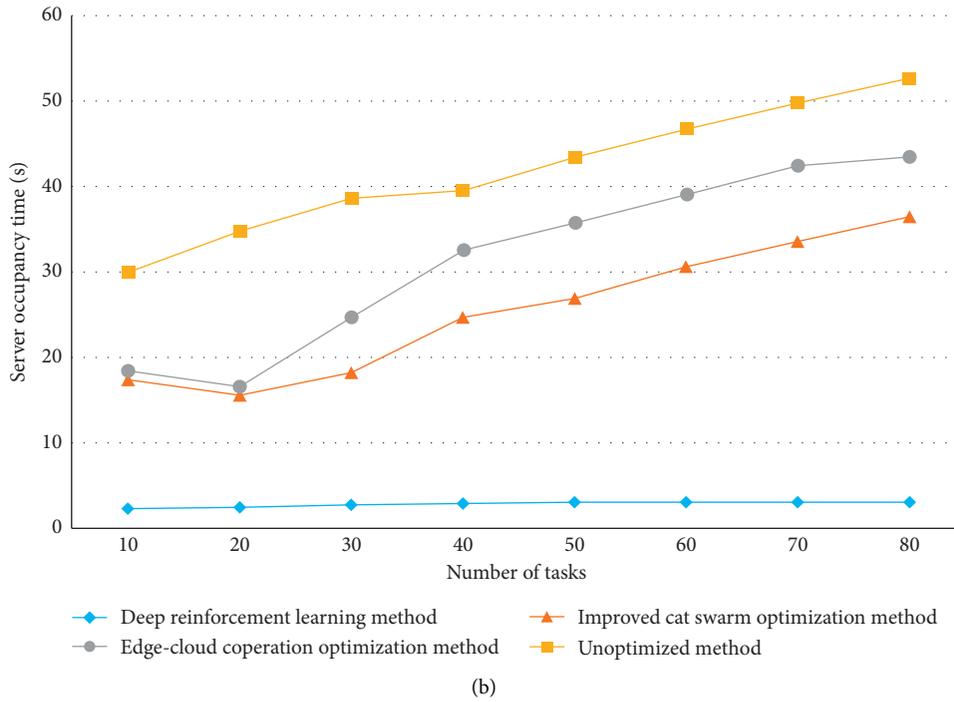
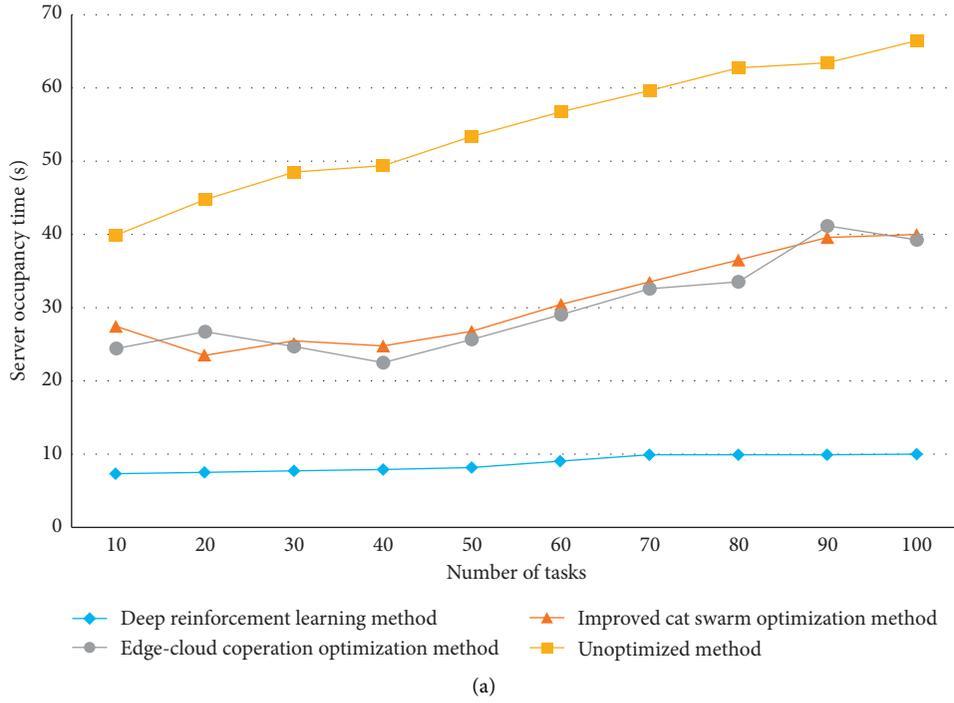


FIGURE 1: Edge server occupancy time. (a) Edge server occupancy time with all edge nodes working. (b) Edge server occupancy time with partial edge nodes working.

TABLE 2: Server power consumption by different methods (W).

Operation hours (H)	Deep reinforcement learning	Improved cat swarm	Edge-cloud cooperation
0.5	4	24	23
1	6	41	38
1.5	8	60	57
2	9	80	78

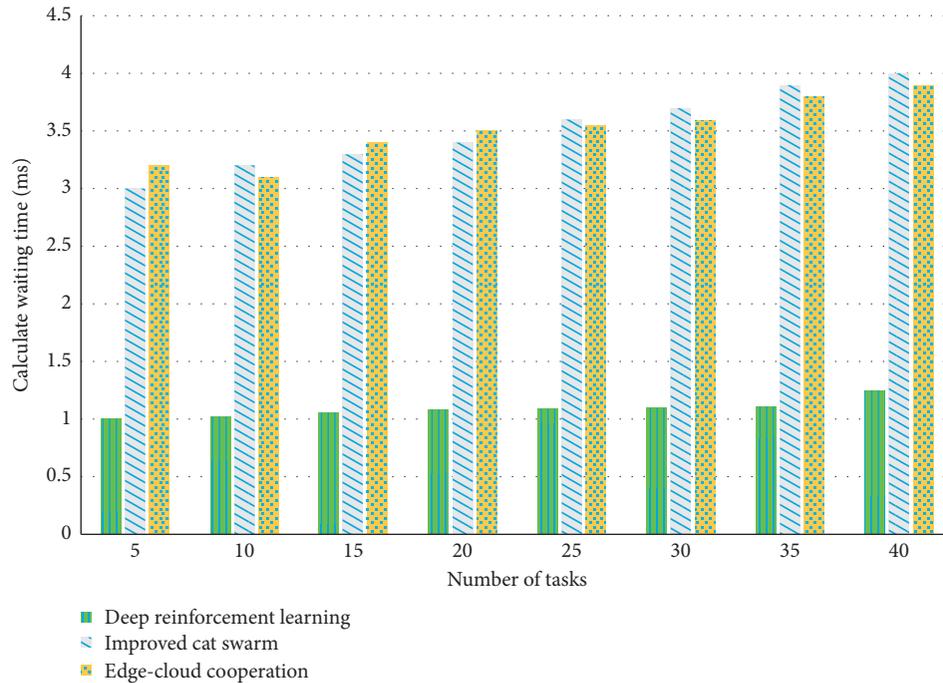


FIGURE 2: Average calculation waiting time under the same method.

5. Conclusions

Physical expansion training has always been a topic of discussion. As an important part of the field of AI, deep learning technology has been strongly sought after by academia and industry in recent years, and the emergence of EC technology corresponding to cloud computing has once again attracted great attention from the academic community. In this paper, the CNN model in deep reinforcement learning technology is used to realize the EC resource allocation of the IoT, and the convolution processing is mainly used to complete the rational allocation of resources. This paper combines AI and EC to propose a method for EC to drive real-time deep reinforcement learning. This method proposed in this paper is used on the wearable sensor device of each trainer in physical expansion training. Through experimental analysis, the method proposed in this paper has low occupancy time, high computing efficiency, and strong applicability; server power consumption is small; and it can effectively control the calculation cost, complete the efficient EC process so that each trainer's training model can be applied to training faster, and improve the quality and accuracy of the training, so as to more effectively carry out physical expansion training. In future research, further optimizations will be made on how to more effectively complete EC, while ensuring the reliability and timeliness of task processing and balancing the load of each edge server during peak periods.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The author declares no conflicts of interest.

References

- [1] Q. Yan, T. Liao, and Y.-j. Zhang, "Digital-based performance training: concepts, advances and applications," *China Sport Science*, vol. 38, no. 11, pp. 3–16, 2018.
- [2] Y. Cao and L. Zheng, "The values, difficulties and countermeasures of artificial intelligence applied in sports field," *Sports Culture Guide*, no. 11, pp. 31–35, 2018.
- [3] J. Ge, "Research on the application of artificial intelligence and big data in sports," *Sport Science And Technology*, vol. 41, no. 2, pp. 30–31, 2020.
- [4] W. Ma and M. A. Jing, "Application and prospect of artificial intelligence in physical education," *Journal of Sports Adult Education*, vol. 36, no. 6, pp. 42–45, 2020.
- [5] L. Lu and X. Li, "Clustering and evolution of artificial intelligence technology in the field of international sports," *Journal of Shandong Institute of Physical Education and Sports*, vol. 36, no. 3, pp. 21–32, 2020.
- [6] Q. Liu, "Network big data technology exploration based on cloud computing and IoT," *Modern Industrial Economy and Informationization*, vol. 11, no. 2, pp. 101–102, 2021.
- [7] Y. Chen, "Research on IoT technology based on big data," *Wuxian Hulian Keji*, vol. 16, no. 13, pp. 27–28, 2019.
- [8] H. Gao and X. h. Li, "Research on cloud edge task distribution strategy in EC," *Information Technology*, no. 2, pp. 103–108, 2021.
- [9] F. N. Iandola, S. Han, M. W. Moskewicz et al., "SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <0.5 MB model size," 2016, <https://arxiv.org/abs/1602.07360>.
- [10] J. X. Wu, L. Cong, Y. H. Wang et al., "Quantized convolutional neural networks for mobile devices," in *Proceedings of the IEEE Conference on Computer Vision and Pattern*

- Recognition (CVPR) 2016*, pp. 4820–4828, IEEE Press, Las Vegas, NV, USA, June 2016.
- [11] K. Kumar and Y. H. Lu, “Cloud computing for mobile users: can offloading computation save energy?” *Computer*, vol. 43, no. 4, pp. 51–56, 2010.
- [12] J. Xu, L. Chen, and S. Ren, “Online learning for offloading and autoscaling in energy harvesting mobile edge computing,” *IEEE Transactions on Cognitive Communications and Networking*, vol. 3, no. 3, pp. 361–373, 2017.
- [13] Y. Mao, J. Zhang, and K. B. Letaief, “Dynamic computation offloading for mobile-edge computing with energy harvesting devices,” *IEEE Journal on Selected Areas in Communications*, vol. 34, no. 12, pp. 3590–3605, 2016.
- [14] J. Xu, L. Chen, and P. Zhou, “Joint service caching and task offloading for mobile EC in dense networks,” in *Proceedings of the IEEE INFOCOM 2018-IEEE Conference on Computer Communications*, pp. 207–215, IEEE, Honolulu, HI, USA, April 2018.
- [15] S. Wang, K. Chan, I.-H. Hou et al., “Red/LeD: an asymptotically optimal and scalable online algorithm for service caching at the edge,” *IEEE Journal on Selected Areas in Communications*, vol. 36, no. 8, pp. 1857–1870, 2018.
- [16] A. Javed, K. Heljanko, A. Buda et al., “Cefiot: a fault-tolerant IoT architecture for edge and cloud,” in *Proceedings of the 2018 IEEE 4th World Forum on IoT (WF-IoT)*, pp. 813–818, IEEE, Singapore, February 2018.
- [17] J. Dean, G. Corrado, R. Monga et al., “Large scale distributed deep networks,” *Advances in Neural Information Processing Systems*, pp. 1223–1231, 2012.
- [18] W. Zhang, S. Gupta, X. Lian et al., “Staleness-aware async-sgd for distributed deep learning,” in *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence*, pp. 2350–2356, AAAI Press, New York, NY, USA, July 2016.
- [19] M. Dai and Y. Huang, “Analysis of physical ability data based on data mining,” *Computer Engineering and Applications*, vol. 39, no. 9, pp. 38–40, 2003.
- [20] S. Huo, B. Li, Z. Zhao, J. Li, and W. Qu, “Motion data acquisition and transmission based on ZigBee module,” *Microcontrollers & Embedded Systems*, no. 12, pp. 41–42, 2009.
- [21] T. Lan, Y. Wang, W. Yang, S. N. Robinovitch, and G. Mori, “Discriminative latent models for recognizing contextual group Activities,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 8, pp. 1549–1562, 2012.
- [22] T. Wang, Y. Zhang, and J. Wang, “Characteristics of time variation in energy metabolism during exercise with moderate intensity,” *Journal of Zhejiang Normal University (Natural Sciences)*, no. 4, pp. 470–474, 2014.
- [23] Y. Xiao, “Sports video classification method based on deep learning,” *Electronic Design Engineering*, vol. 29, no. 3, pp. 162–166, 2021.
- [24] Z. Li, “Research on sports video classification based on deep learning and transfer learning,” *Electronic Measurement Technology*, vol. 43, no. 18, pp. 21–25, 2020.
- [25] L. Kong, R. Wang, N. Zhang, and H. Li, “Survey on AI detection and recognition algorithms based on EC,” *Radio Communications Technology*, vol. 45, no. 5, pp. 453–462, 2019.
- [26] R. Cai, C. Zhong, Y. Yu, B. Chen, Y. Lu, and Y. Chen, “CNN quantization and compression strategy for EC applications,” *Journal of Computer Applications*, vol. 38, no. 9, pp. 2449–2454, 2018.