

## *Retraction*

# **Retracted: Design and Research of the AI Badminton Model Based on the Deep Learning Neural Network**

### **Journal of Mathematics**

Received 23 January 2024; Accepted 23 January 2024; Published 24 January 2024

Copyright © 2024 Journal of Mathematics. 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.

This article has been retracted by Hindawi following an investigation undertaken by the publisher [1]. This investigation has uncovered evidence of one or more of the following indicators of systematic manipulation of the publication process:

- (1) Discrepancies in scope
- (2) Discrepancies in the description of the research reported
- (3) Discrepancies between the availability of data and the research described
- (4) Inappropriate citations
- (5) Incoherent, meaningless and/or irrelevant content included in the article
- (6) Manipulated or compromised peer review

The presence of these indicators undermines our confidence in the integrity of the article's content and we cannot, therefore, vouch for its reliability. Please note that this notice is intended solely to alert readers that the content of this article is unreliable. We have not investigated whether authors were aware of or involved in the systematic manipulation of the publication process.

Wiley and Hindawi regrets that the usual quality checks did not identify these issues before publication and have since put additional measures in place to safeguard research integrity.

We wish to credit our own Research Integrity and Research Publishing teams and anonymous and named external researchers and research integrity experts for contributing to this investigation.

The corresponding author, as the representative of all authors, has been given the opportunity to register their agreement or disagreement to this retraction. We have kept a record of any response received.

### **References**

- [1] Y. Chen and H. Hu, "Design and Research of the AI Badminton Model Based on the Deep Learning Neural Network," *Journal of Mathematics*, vol. 2022, Article ID 6739952, 10 pages, 2022.

## Research Article

# Design and Research of the AI Badminton Model Based on the Deep Learning Neural Network

Yujue Chen<sup>1</sup> and He Hu <sup>2</sup>

<sup>1</sup>College of Physical Education and Sport Academy, Qinghai Normal University, Qinghai, Xining 810008, China

<sup>2</sup>College of Computer Science and Technology, Xi'an University of Science and Technology, Shanxi, Xian 710000, China

Correspondence should be addressed to He Hu; huhepingzi@163.com

Received 2 December 2021; Revised 19 December 2021; Accepted 22 December 2021; Published 20 January 2022

Academic Editor: Naeem Jan

Copyright © 2022 Yujue Chen and He Hu. 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.

In view of the fact that it is difficult for existing algorithms to identify the movements of a player in an accurate way, this paper puts forward an artificial intelligence (AI) motion model on the basis of the deep learning neural network instruction set architecture (ISA). Firstly, a mobile neural network (MNN) inference engine was utilized to create a new AI sports project-side intelligent practice model. Under this model, a movement can be segmented into a series of decomposition movements, which are recognized and judged separately for the purpose of measuring the entire movement. In order to test its feasibility, the study compares the MNN inference engine with the traditional reasoning engine in terms of their algorithmic capabilities and compares the results obtained through this algorithm and traditional online motion app. Research shows that, in the MNN of the AI sports project proposed in this paper, the datasets of action recognition exceed the results of other inference engines, characterized by lightweight, high performance, and accessibility. Research also demonstrates that the AI sports project model can adapt to the needs of sports projects with a variety of themes and improve the accuracy of movement recognition details.

## 1. Introduction

In the sustainable exploration of end intelligence, practice and service empowerment take place in the scenario of healthy life with sports, namely, sports AI project [1]. Projects of this kind contributed greatly to the realization of the core goal of sports digitization and the steady growth of China's sports population, thus becoming a crucial step in intelligence sports [2]. The outbreak of the COVID-19 pandemic has added to the difficulty of traditional offline sports and promoted the development of home-based sports in the context of AI technology. Through technological precipitation, home sports are combined with online sports and further empowered by AI technology, hence intelligent sports or AI sports. With intelligence and movement as their core concept, AI sports are complex that are composed of the process and the conclusion of academic activities [3, 4]. Their ultimate purpose is to create a new, simple, and interesting way for users to exercise at home [5]. To elaborate,

one merely needs a mobile phone or a few-square-meter field to do AI sports. As far as badminton is concerned, the user just needs to open the power app, position the phone on the side of the playing field in an appropriate angle, and adjust the phone-subject distance according to the app's automatic voice prompts until the entire image of the player is incorporated in the recognition frame [6].

AI is a marvelous helper for physical training in that it can detect digital features and associations that are not easy to discover through human eyes or the brain, thus making AI sports more exploratory [7, 8]. This paper focuses on the integration of the badminton with AI intelligence and movements and aims to remove mistakes in beginners' movements in a more targeted way and to improve their skills in playing badminton. Firstly, the mobile neural network (MNN) inference engine is utilized to digitize the sport and obtain experimental data. Secondly, a preliminary analysis is made concerning the differences between the data and those obtained through the traditional sports app.

Thirdly, the outcomes show how the badminton players should improve their movements in playing badminton in terms of accuracy, comprehensiveness, movement coordination, and partner cooperation and make their movements approximate professional players. Fourthly, some suggestions are provided as to the application of AI in sports. Moreover, in this study, we first defined the main conceptions and their theoretical foundation in Section 2. Then, we carried out a discussion on the technical supports such as AI and intelligent motion in Section 3. Furthermore, the accuracy is recognized, the consumption of performance is decreased, and the improvement in the efficiency is tested. In addition, the flow of the model of AI sports is given in Section 4 of the paper. In Section 5, the comparative analysis is carried out where we performed the experiments for the stated purpose. Finally, the conclusion of the paper is given in Section 6.

## 2. Core Concept Definition and Theoretical Foundation

The inference forecast performed in the cloud can then be attempted on the mobile as mobile computing power and deep learning are undergoing rapid development, and small-scale network models are getting all the more mature. The end-side intelligence deploys and runs the AI algorithm at the end. In comparison to server-side intelligence, end-side intelligence has the advantages of low latency, good balance of data privacy, and cloud savings [9, 10].

MNN is a lightweight inference engine based on the deep neural network (DNN), and loaded on its end side are DNN models. Up to now, MNN has seen wide applications in face detection, gesture recognition, portrait segmentation, and other things [11].

While traditional sports are one-way streamlined, AI sports project is a client-side sports intelligence system that can verify DEMO, systematize various dynamics, and support technical capability transformation. The intelligent sports system realizes terminal inference of the mobile phone through the deep inference engine [12]. Information response and dynamic corrections are made by identifying and analyzing postures and dynamics, sports track, and dynamic angles, and then, the modular combination of technical capabilities is carried out. Currently, the system, which is characterized by an organic integration of sports and AI technology, supports over ten kinds of sports actions and dozens of ways of playing, making online badminton more simple.

## 3. Technical Supports

The primary technical idea of intelligent motion at the AI sports end is to use the MNN inference engine for reasoning and pose recognition. It includes the following:

- (1) Measuring real-time body contour in pictures and videos and finding fourteen important bone points and essential joint parts such as the head, arms, and feet

- (2) Connecting the points to form movement images and analyzing body posture, movement angle, and trajectory

- (3) Measuring the user's badminton action via the action and posture matching system and carrying out dynamic timing and counting

At the same time, through real-time monitoring and analysis of the standard action, response and interaction are made to improve user experience and interaction.

In traditional badminton training, people can get timely guidance and support from on-site assistants, such as coaches, examiners, relatives, and friends. Smart fitness programs on the AI sports side, nevertheless, allow people to interact directly with mobile apps while making movements. The ability and cognitive level of human-computer interaction will be affected by various factors, including logical reasoning modelling ability, badminton scene complexity, and sports information matching recognition calculation. Some new problems and difficulties appear in the study and execution of remote intelligent human locomotion ability, such as human-computer location matching, insufficiency in bone point recognition, mistakes in joint point identification, two-dimensional distortion, irrelevant user movement, mobile phone shaking, and scene noise.

Badminton dynamic effect evaluation and key algorithm design are conducive to the improvement of matching accuracy of action nodes [13], which is regarded as the cornerstone of human movement. On the premise of enhancing identification efficiency, corresponding measures should be taken to reduce resource consumption of mobile terminals, which are mainly manifested in battery power and heat generation, and improve user experience. In this way, manpower and time consumption involved in mobile terminal testing will be reduced. Additionally, the efficiency of R&D and testing is supposed to be raised to provide strong support for smooth and effective interaction within the team.

**3.1. Accuracy Recognition.** The most direct and primary exercise experience for users of intelligent badminton is the accuracy of dynamic counting [14]. Once there is a counting error in action recognition, sports users' enthusiasm in using the app will be affected, and their participation initiative will be demotivated. For this reason, counting problems should be avoided in the first place.

The basic principle of the intelligent motion calculation is to decompose an overall action into several decomposition movements and then employ different processes for movement recognition and judgment [15]. After a set of operations is completed, the effects of all dynamics are judged. If it is effective, the count is increased by 1. Otherwise, the steps will be repeated. In short, the recognition and operation in intelligent motion are all state machines. A motion action is discretized and abstracted into  $N$  state machines  $\{S(0), S(1), S(2) \dots, S(n1)\}$ , which are tested successively. When all state machines are detected, the user has completed the action; then, 1 is added to the calculation

result. In case that a certain state machine is not detected, the system will give feedback and reset the machine. Each condition machine corresponds to a specific automatic trigger condition. The following dynamic matching result can be obtained by detecting the cyclic correspondence between the real-time skeletal point position change and the current state, thus enhancing the stability of the skeletal point and ensuring outcome accuracy. Since the dynamic identification accuracy is closely related to the dynamic collocation calculation, the better the calculation collocation effect, the higher the identification accuracy. To improve the accuracy of badminton motion recognition, the main factors affecting the calculation of motion information matching, such as the skeletal point, state machine, and matching, can be chosen as point cuts. Specific methods are as follows:

- (1) Select the action with stable, easily recognizable, and iconic bone points as the state machine.
- (2) The frame rate should be able to cover all the state machines of a badminton motion. The accuracy of bone point recognition has a great impact on motion matching.

As is shown in Figures 1 and 2, when an error occurs in the identification of the left arm point, a straight match will get an erroneous result. In this case, it is necessary to utilize the dynamic historical information of badminton users to adjust the dynamic matching result using the dynamic matching algorithm.

**3.2. Performance Consumption Reduction.** Restricted by physical conditions, the computing power and storage space on the cell phone are limited. Furthermore, deep learning reasoning requires enormous data consumption due to the great deal of complex calculating work. In-depth learning and reasoning will take up a lot of resources of the mobile terminal and CPU, and memory consumption will be increased significantly, leading to overheat and battery over-consumption for the equipment. Therefore, when the intelligent motion device runs on the mobile terminal, performance loss must be avoided so as to improve user experience.

To effectively reduce performance loss of the overall system, it is essential to reduce loss in every step, as is shown in the chart in Figure 3.

These three phases perform different functions. Before logical reasoning, the format conversion is performed; that is, the stream data signals acquired by the camera are converted into various stream data information formats such as YUV format and RGBA format required by the reasoning process (Figure 4). In the inference stage, the input and output skeletal point positions are calculated. The inference engine can perform a series of operations on the input frame number to draw relevant logical inference conclusions. For example, attitude recognition is to convert the RGBA data information of the input image into information of bone point positions. The postreasoning stage involves a series of analyses concerning the performance,



FIGURE 1: Nodes of swing action.

rendering operations and business-related actions, such as UI display and animation effect display.

The above three stages can be optimized as follows. The optimization of the inference process can be completed by the deep inference engine MNN. Stream data in the pre-reasoning stage can be converted directly into the required format, without having to rely on intermediate transformation. Raw data in the reasoning stage can be directly converted into the RGBA format, thus reducing unnecessary calculations and alleviating the burden for the terminal. In addition, appropriate rendering methods, such as multi-backend abstraction and mixed scheduling, should be selected for the platform bearing the postinference stage to reduce rendering loss. For the IOS platform, metal can be used directly for rendering enhancement.

**3.3. Testing Efficiency Improvement.** AI intelligent sport is a bold attempt in digitizing sports [16]. Its R&D, particularly the testing process, requires a large amount of investment in terms of time, equipment, and effort to improve the application from various aspects. In addition, the effective detection method for AI motion recognition is greatly influenced by environmental factors, such as light source,

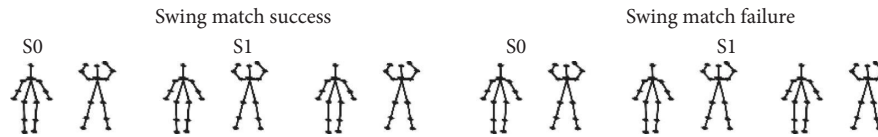


FIGURE 2: Matching diagram of action nodes.

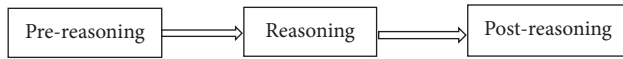


FIGURE 3: MNN reasoning diagram.

background, motion distance, and the size of a person's image in a shot, putting effective detection methods to the test.

Take the traditional badminton test method as an example. Generally, the detection person has to firstly manually record the real-time actions of real people on the site and make analyses afterwards off the site, as is shown in Figure 5.

In view of the fact that different brands of phones vary in drivers, operating systems, and specific performance parameters, etc., it is quite difficult to take all factors into account when traditional detection methods are employed. This poses great challenges to testers and cannot guarantee detection uniformity and accuracy at the same time. Specific reasons are as follows [17]:

- (1) The labour costs are high: a test requires the cooperation of several students, which is time-consuming and exhausting.
- (2) The test environment is relatively homogeneous: it cannot adapt to the complex and changing environment on the route.
- (3) It is difficult to quantify the test results. It is impossible to quantitatively evaluate model performance, calculation validity, matching accuracy and precision, resource consumption, etc.
- (4) Problem location is difficult. The postanalysis and troubleshooting fail to respond to online customer complaints in a targeted way.

The traditional badminton action node testing cannot solve these problems. For this purpose, Shanghai Sports Science and Technology Group has developed an AI sports automatic testing tool and solved the problems commonly found in traditional testing methods. It has realized rapid positioning and regressing badminton nodes online and quantitatively evaluating the calculation accuracy of a model.

The basic processing idea of the automatic testing tool is to simulate the actual situation through batch analysis of video sets, collect bone point data (Figure 6), complete the detection of business results, and automatically form a test report. The specific technical methods are demonstrated in Figure 7.

ISA is an unsupervised learning method with a two-layer network generation model that can effectively simulate the hierarchical response model of simple cells' and complex

cells' receptive fields in the V1 region of the human visual system [18]. The most basic implementation method of ISA is to use the first layer of the model to learn the weight  $W$  of a linear transformation (L1 is similar to FC) [19, 20]. Next, the same subspace elements are combined in the second layer. Then, a fixed nonlinear transformation  $V$  (L2 pooling) is performed to obtain features that are invariant in response to phase changes.

With the introduction of the latest AI testing tool developed by ISA which is introduced, the labour cost is significantly reduced, and the detection performance is greatly improved. It is noteworthy that the effect of the test tool is related to the number of samples tested. The more abundant the models, the better the detection accuracy.

#### 4. The Model Flow of AI Sports

The flow of the model in the diagram in the figure is the overall processing on a mobile terminal of the rear-facing camera on IOS, or Android devices take a front-facing shoot (Figure 8).

First, the system gets the data from the camera as an input to the SDK. Then, the SDK performs the following operations.

Before the MNN engine performs the inference, the original input is processed to guarantee that the face inference in the input data is forward using the AI model. The results are generated based on the key points input to the image coordinate system after preprocessing, and the critical point coordinates are transformed to the same direction as the screen rendering coordinate system to facilitate rendering. The ultimate key points are displayed on the user's screen in the process application, and the front end uses a "canvas" for rendering. The coordinate system of the canvas is called the rendering coordinate system. In the last step of SDK detection, we transform the critical points to the same orientation as the rendering coordinate system and then map the key point coordinates to the coordinates of the rendering coordinate system at an equal scale. After mapping is completed, the results are directly rendered to the canvas.

#### 5. Project Experiment and Comparative Analysis

A comparison between MNN and TVM algorithms is made to test the feasibility of the proposed AI sports with the MNN inference engine.

TVM owns fully automatic search through ML, while the MNN is semiautomatic. It is the biggest and the fatal drawback in terms of refinement and optimization (Figures 9–11).

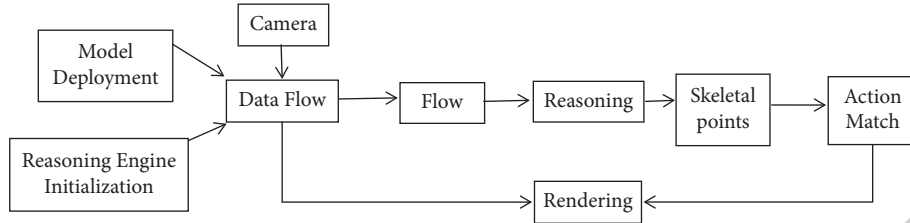


FIGURE 4: MNN process.

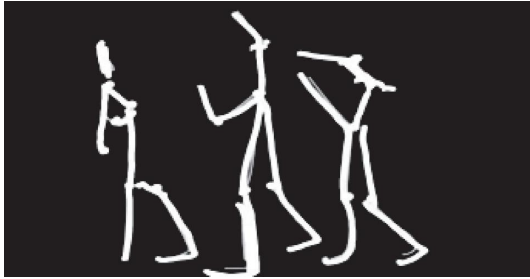


FIGURE 5: The action node of striking lofty goals of badminton.

### 5.1. Preference

**5.1.1. Accelerated Scheme Selection.** In mobile applications, computation speed and lightness are the primary considerations. Acceleration libraries, such as OpenBLAS [21] and Eigen, cannot be used in mobile applications with a view to alleviating operation burden of the terminal. Therefore, NCNN (Tencent, 2017), MACE (Xiaomi, 2018), and Anakin (Baidu, 2018) opt for a manual search approach that does not rely on any external libraries and implements operators using assembly instructions case by case. This approach makes the reasoning engine lightweight and efficient, but the case-by-case optimization is also time-consuming and difficult to cover for all operators.

Fully automated search is in sharp contrast to manual search. The typical representative is TVM, which solves the problem of redundant dependencies and provides graph-level and operator-level optimizations for both the model and the back end [22]. Hence, TVM has excellent support for the model and device diversity. However, it comes at a cost. The runtime library generated by TVM is model-specific. In other words, when the model needs to be updated, TVM is required to regenerate the runtime library, which is unacceptable for mobile applications. MNN adopts a semiautomatic search approach with enhanced generality and performance.

**5.1.2. Calculation Scheme Selection.** MNN operates on a cost evaluation mechanism which takes algorithm implementation and backend characteristics into full account so as to find the optimal solution. The following is the cost calculation formula:

$$C_{\text{total}} = C_{\text{algorithm}} + C_{\text{backend}}. \quad (1)$$

To minimize the overall costs, it is crucial to opt for the fastest algorithm and the most efficient backend.

Convolution is taken as an example to elucidate the cost of algorithms. There are currently two fastest implementation algorithms—sliding window and Winograd. For a variety of convolution configurations, the algorithm with the lowest computational cost is selected in a dynamic way. The selection method is as follows:

- (1) If the kernel size is 1 in a matrix multiplication, the Strassen algorithm is the most appropriate.
- (2) If the kernel size exceeds 1, Winograd is recruited to transform the convolution operation into matrix multiplication. Theoretically, the cost of convolution can be expressed by the following formula:

$$C_{(n)} = 2i_c(n+k-1)^3 + i_c o_c(n+k-1)^2 + n(n+k-1)(2n+k-1), \quad (2)$$

$$n = \arg \min C_{(n)}. \quad (3)$$

Based on formulae (2) and (3), the optimal output size can be chosen to minimize the cost. So, the cost for convolution is evaluated as follows.

Scheme = sliding window: if  $k > 1$  and  $n = 1$ ,

$$F(n \times n, k \times k) \quad \text{if } k > 1 \text{ and } n > 1. \quad (4)$$

The second problem is how to calculate and minimize the backend cost. That is, the best backend is selected for each operator to ensure the lowest global costs:

$$C_{\text{backend}} = \sum C_{\text{op}}. \quad (5)$$

**5.2. Preparation-Execution Decoupling.** During the execution of the program, the calculation is typically accompanied by memory requests and releases. For mobile applications, the overhead in memory management is considerable. Considering that the input size is already determined, the engine can execute all the operators virtually to meet the exact memory requirements. In this way, the required memory can be allocated in advance during the preference stage and reused during the execution stage. The principle is shown in Figure 12.

**5.3. Kernel Optimization.** A kernel refers to the detailed implementation of an operator [23]. The optimization comes from two primary sources: algorithms and scheduling, i.e., choosing the algorithm with the lowest complexity and taking good advantage of hardware resources.

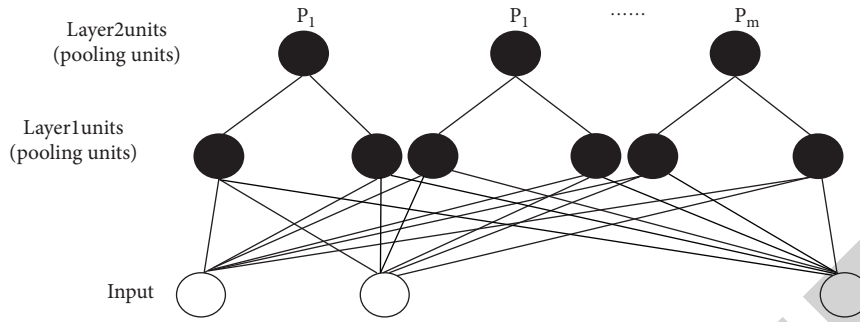


FIGURE 6: ISA neural network instruction diagram.

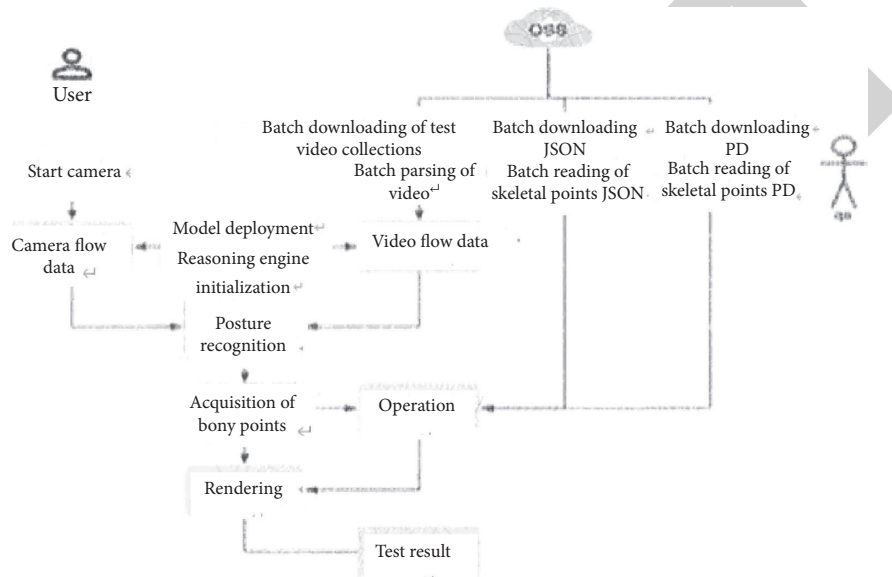


FIGURE 7: Automatic test flow.

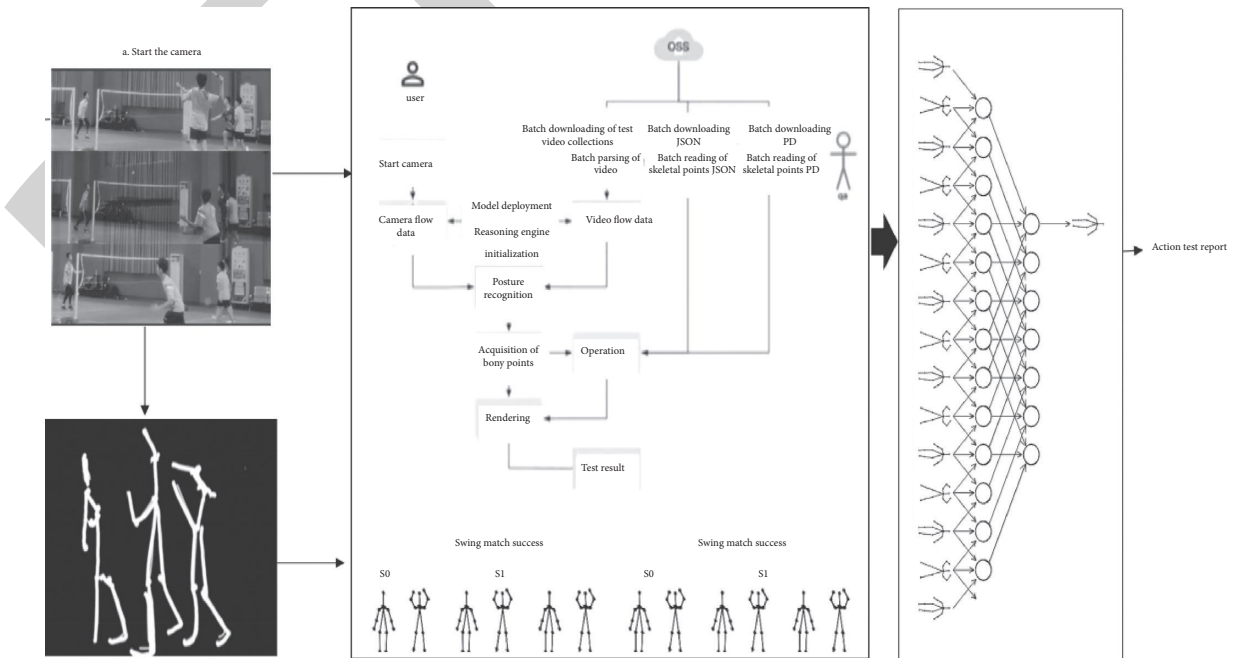


FIGURE 8: AI sports model.

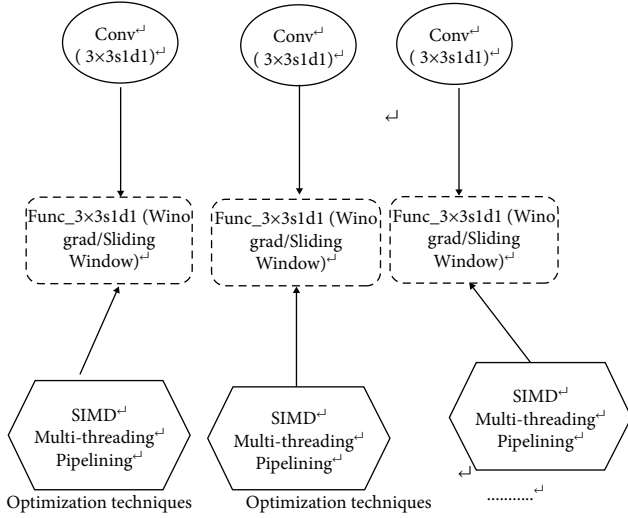


FIGURE 9: Manual search.

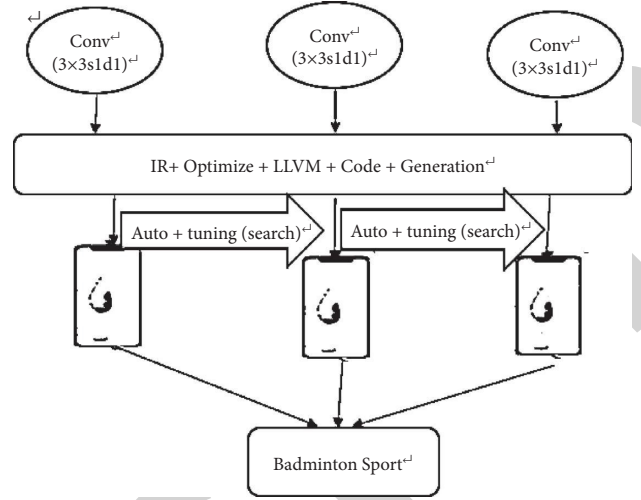


FIGURE 11: Automatic search (TVM).

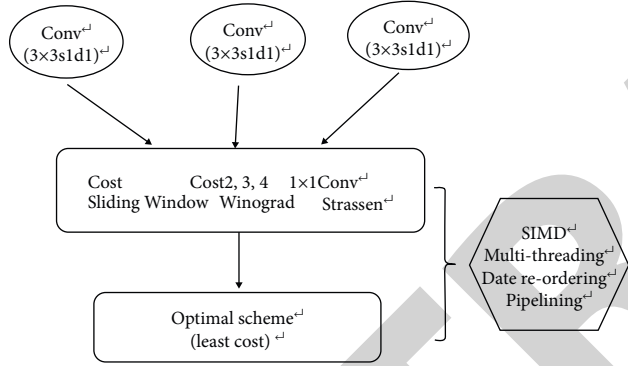


FIGURE 10: Semiautomatic search (MNN).

5.3.1. *Winograd Optimization.* The Winograd-based fast convolution algorithm has been widely applied in numerous reasoning frameworks. Different search methods are compared through three processes: (a) the manual search can be optimized by continuous correction, which means that the operators have to perform a case-by-case optimization and error correction. (b) Then, the semiautomatic MNN searches for the optimal action to match in high-performance computing. (c) Finally, the automatic search (TVM) matches the correct action for compiler optimizations through automatic filtering throughout. The Winograd optimization of data outflow in semiautomatic search is as follows:

$$F(2, 3) = \begin{bmatrix} d_0 & d_1 & d_2 \\ d_1 & d_2 & d_3 \end{bmatrix} \begin{bmatrix} g_0 \\ g_1 \\ g_2 \end{bmatrix} = \begin{bmatrix} r_0 \\ r_1 \end{bmatrix}. \quad (6)$$

$$\begin{aligned} \Rightarrow r_0 &= (d_0 \cdot g_0)(d_1 \cdot g_1)(d_2 \cdot g_2). \\ r_1 &= (d_1 \cdot g_0)(d_2 \cdot g_1)(d_3 \cdot g_2). \end{aligned}$$

$$F(2, 3) = \begin{bmatrix} d_0 & d_1 & d_2 \\ d_1 & d_2 & d_3 \end{bmatrix} \begin{bmatrix} g_0 \\ g_1 \\ g_2 \end{bmatrix} = \begin{bmatrix} m_1 + m_2 + m_3 \\ m_2 - m_3 - m_4 \end{bmatrix}. \quad (7)$$

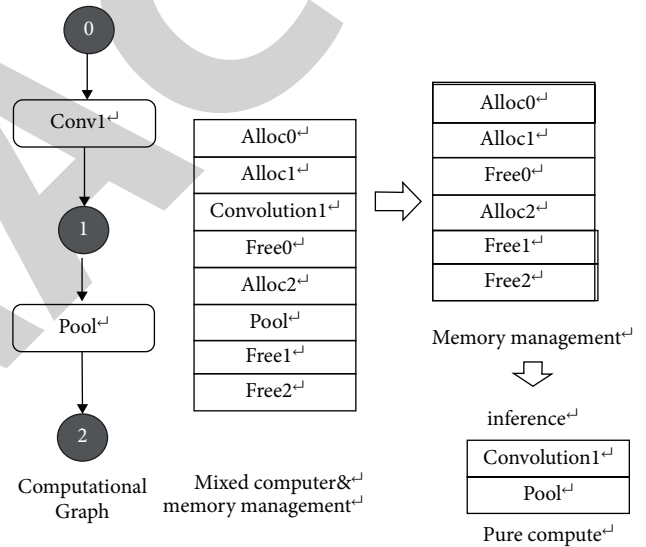


FIGURE 12: The preinference principle diagram.

$$\Rightarrow m_1 = (d_0 - d_2)g_0 + g_1 + g_2/2.$$

$$m_4 = (d_1 - d_3)g_0 - g_1 + g_2/2.$$

However, Strassen ( $O(n^3) \rightarrow O(n \log^2 7)$ )6.

$$\begin{bmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{bmatrix} = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \cdot \begin{bmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{bmatrix}. \quad (8)$$

- S1 = B12-B22
- S2 = A11 + A12
- S3 = A21 + A22
- S4 = B21-B11
- S5 = A11 + A22
- S6 = B11 + B22
- S7 = A12-A22
- S8 = B21 + B22



TABLE 1: Comparison results of different inference engines.

| Mobile inference engine | CPU | GPU   |        |        | Supported OS  |
|-------------------------|-----|-------|--------|--------|---------------|
|                         |     | Metal | OpenGL | OpenGL |               |
| CoreML                  | 110 | 110   | —      | —      | IOS           |
| TF-Lite                 | 92  | 15    | —      | —      | IOS + Android |
| MACE                    | 62  | —     | —      | 29     | Android       |
| NCNN                    | 67  | —     | —      | —      | IOS + Android |
| MNN                     | 93  | 55    | 17     | 32     | IOS + Android |

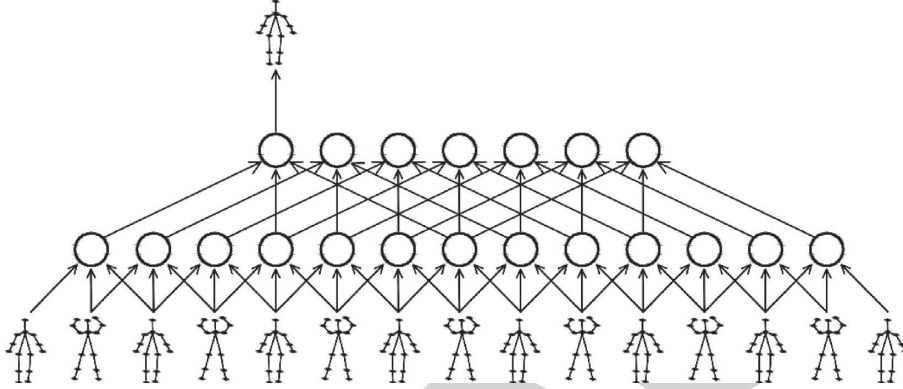


FIGURE 13: Neural network diagram of the ISA model.

TABLE 2: Comparison of ISA model results with different command models.

|   | KTH (%) | UCF (%) | Sports (%) | Hollywood 2 (%) |
|---|---------|---------|------------|-----------------|
| Best results for other instruction models | 92.73   | 50.3    | 85.8       | 71.6            |
| AI sports results                         | 93.9    | 53.3    | 86.5       | 75.8            |

$$S9 = A11 - A21$$

$$S10 = B11 + B12$$



$$P1 = A11 \cdot S1 = A11 \cdot B12 - A11 \cdot B22$$

$$P2 = S2 \cdot B22 = A11 \cdot B22 + A12 \cdot B22$$

$$P3 = S3 \cdot B11 = A21 \cdot B11 + A22 \cdot B11$$

$$P4 = A22 \cdot S4 = A22 \cdot B21 - A22 \cdot B11$$

$$P5 = S5 \cdot S6 = A11 \cdot B11 + A11 \cdot B22 + A22 \cdot B11 + A22 \cdot B22$$

$$P6 = S7 \cdot S8 = A12 \cdot B21 + A12 \cdot B22 - A22 \cdot B21 - A22 \cdot B22$$

$$P7 = S9 \cdot S10 = A11 \cdot B11 + A11 \cdot B12$$

$$-A21 \cdot B11 - A21 \cdot B12$$



$$C_{11} = P5 + P4 - P2 + P6$$

$$C_{12} = P1 + P2$$

$$C_{21} = P3 + P4$$

$$C_{22} = P5 + P1 - P3 - P7$$

Operator convolution and large-scale matrix multiplication optimization are mainly embodied in the application of two classical algorithms [24, 25]. Many inference

frameworks using Winograd are hard-coded, i.e., the three matrices corresponding to the kernel and input sizes are determined, making the scalability poor in the face of new scenarios. On the contrary, the Winograd generator enables Winograd to adapt to arbitrary kernel and input size.

In addition, the Strassen algorithm is used in the MNN to optimize matrix multiplication. MNN is the first mobile inference engine using Strassen algorithm to optimize large matrix multiplication. Strassen replaces several multiplication operations with addition operations. In general, the processor performs the addition operations much faster than the multiplication operations, thus causing a speedup effect. This speedup can be maximized using recursive calls, which requires determining the conditions for the end of the recursion (Table 1).

Furthermore, MNN supports major mobile data devices and has the function setting of hybrid scheduling, which solves the troubles caused by repeated scheduling and facilitates lightweight.

As is shown in Figure 13, a video sequence of the swing action is compressed into a vector as the input value. In AI sports projects, the output of the ISA model is combined as the final output vector to improve the accuracy of movement recognition.

- (1) The first layer of the model learns the weight output of the linear transformation.

- (2)  $W$  between the first and second layers is the weights to be learned. The weight  $V$  of the second and third layers of the output layer is fixed and does not need to be learned.

$$p_i(x^t; W, V) = \sqrt{\sum_{k=1}^m V_{ik} \left( \sum_{j=1}^n W_{kj} x_j^t \right)^2},$$

$$\min_W i \text{ mize } \sum_{t=1}^T \sum_{i=1}^m p_i(x^t; W, V),$$

(9)

subject to  $WW^t = I$ .

The AI sports project is formally based on ISA. And it satisfies the weight  $W$  orthogonal matrix.

From the above experiments (Table 2), it can be concluded that the MNN solves the problem of redundancy, provides graph-level and operator-level optimization for the model and backend, and enhances versatility and performance of the search approach. In addition, the support of the ISA instruction set structure for neural networks makes MNN more mature and applicable in the AI sports field.

## 6. Conclusion

Recently, the sports researchers have shifted their focus to AI which, now, has a wide application in sports management. There are many challenges for applications of sports AI projects in sports because the variation of data of sports events, competitions, and teaching led to insufficient resource supply capacity against the backdrop of rapid updating of mobile devices and AI technology development.

AI sports system now supports dozens of badminton sports. In addition, a large number of AI training and learning courses have been developed. Through the modular integration of sports functions, it will contribute much to the expansion of its services in various aspects of sports in the future.

Since the advent of AI intelligent sports technology to date, upper body movements such as straight arm rope and push-up, torso movements such as hip bridge and deep squat, and systemic movements such as badminton games and singles-doubles have been successively launched in various sports-related apps. This enables online sports users to free from time and place constraints and participate in AI sports at any time and place at their will, thereby enhancing its attractiveness and efficiency for users.

## Data Availability

The data underlying the results presented in this study are available within the manuscript.

## Disclosure

The authors confirm that the content of the manuscript has not been published or submitted for publication elsewhere.

## Conflicts of Interest

The authors declare no conflicts of interest.

## Authors' Contributions

Both authors saw the manuscript and approved to submit to the journal.

## Acknowledgments

This work was supported by the National Natural Science Foundation of China (no. 11551003).

## References

- [1] A. F. Manfredini, A. M. Malagoni, H. Litmanen et al., "Performance and blood monitoring in sports: the artificial intelligence evoking target testing in antidoping (A.R.I.-E.T.T.A.) project," *The Journal of Sports Medicine and Physical Fitness*, vol. 51, pp. 153–159, 2011.
- [2] A. C. Lapham and R. M. Bartlett, "The use of artificial intelligence in the analysis of sports performance: a review of applications in human gait analysis and future directions for sports biomechanics," *Journal of Sports Sciences*, vol. 13, no. 3, pp. 229–237, 1995.
- [3] R. Bartlett, "Artificial intelligence in sports biomechanics: new dawn or false hope?" *Journal of Sports Science & Medicine*, vol. 5, pp. 474–479, 2006.
- [4] P. N. Ramkumar, B. C. Luu, H. S. Haeberle, J. M. Karnuta, B. U. Nwachukwu, and R. J. Williams, "Sports medicine and artificial intelligence: a primer," *The American Journal of Sports Medicine*, Article ID 3635465211008648, 2021.
- [5] G. Prabhu, N. E. O'Connor, and K. Moran, "Recognition and repetition counting for local muscular endurance exercises in exercise-based rehabilitation: a comparative study using artificial intelligence models," *Sensors*, vol. 20, 2020.
- [6] A. C. N. Rodrigues, A. S. Pereira, R. M. S. Mendes, A. G. Araújo, M. S. Couceiro, and A. J. Figueiredo, "Using artificial intelligence for pattern recognition in a sports context," *Sensors*, vol. 20, 2020.
- [7] V. Dhar, "What is the role of artificial intelligence in sports?" *Big Data*, vol. 5, no. 3, pp. 173–174, 2017.
- [8] T. Liu, D. Wilczyńska, M. Lipowski, and Z. Zhao, "Optimization of a sports activity development model using artificial intelligence under new curriculum reform," *International Journal of Environmental Research and Public Health*, vol. 18, 2021.
- [9] J. Yuan, J. Zhang, L. Shen, D. Zhang, W. Yu, and H. Han, "Massive data management and sharing module for connectome reconstruction," *Brain Sciences*, vol. 10, 2020.
- [10] M. R. Schädler, A. Warzybok, S. Hochmuth, and B. Kollmeier, "Matrix sentence intelligibility prediction using an automatic speech recognition system," *International Journal of Audiology*, vol. 54, no. Suppl 2, pp. 100–107, 2015.
- [11] P. Palee, B. Sharp, L. Noriega, N. Sebire, and C. Platt, "Heuristic neural network approach in histological sections detection of hydatidiform mole," *Journal of Medical Imaging*, vol. 6, Article ID 044501, 2019.
- [12] T. Wang and J. Park, "Design and implementation of intelligent sports training system for college students' mental health education," *Frontiers in Psychology*, vol. 12, Article ID 634978, 2021.

- [13] T. Ozmen and M. Aydogmus, "Effect of core strength training on dynamic balance and agility in adolescent badminton players," *Journal of Bodywork and Movement Therapies*, vol. 20, no. 3, pp. 565–570, 2016.
- [14] J. Xie, G. Chen, and S. Liu, "Intelligent badminton training robot in athlete injury prevention under machine learning," *Frontiers in Neurorobotics*, vol. 15, Article ID 621196, 2021.
- [15] T. Roland, "Motion artifact suppression for insulated EMG to control myoelectric prostheses," *Sensors*, vol. 20, 2020.
- [16] G. Kakavas, N. Malliaropoulos, R. Pruna, and N. Maffulli, "Artificial intelligence: a tool for sports trauma prediction," *Injury*, vol. 51, no. Suppl 3, pp. S63–S65, 2020.
- [17] M. Fuchs, O. Faude, M. Wegmann, and T. Meyer, "Critical evaluation of a badminton-specific endurance test," *International Journal of Sports Physiology and Performance*, vol. 9, no. 2, pp. 249–255, 2014.
- [18] A. W. Kraft, A. Mitra, A. Q. Bauer et al., "Visual experience sculpts whole-cortex spontaneous infraslow activity patterns through an arc-dependent mechanism," *Proceedings of the National Academy of Sciences*, vol. 114, no. 46, pp. E9952–E9961, 2017.
- [19] M. S. Hosseini, Y. Zhang, and K. N. Plataniotis, "Encoding visual sensitivity by MaxPol convolution filters for image sharpness assessment," *IEEE Transactions on Image Processing*, vol. 28, no. 9, 2019.
- [20] R. Driscoll, Y. Page, S. Lassarre, and J. Ehrlich, "LAVIA—an evaluation of the potential safety benefits of the French intelligent speed adaptation project," *Annual Proceedings*, vol. 51, pp. 485–505, 2007.
- [21] T. Blattner, W. Keyrouz, S. S. Bhattacharyya, M. Halem, and M. Brady, "A hybrid task graph scheduler for high performance image processing workflows," *Journal of Signal Processing Systems*, vol. 89, no. 3, pp. 457–467, 2017.
- [22] A. Janowczyk, S. Chandran, R. Singh et al., "Hierarchical normalized cuts: unsupervised segmentation of vascular biomarkers from ovarian cancer tissue microarrays," *Medical Image Computing and Computer-Assisted Intervention—MICCAI 2009*, vol. 12, pp. 230–238, 2009.
- [23] S. Klus, F. Nüske, and B. Hamzi, "Kernel-based approximation of the Koopman generator and Schrödinger operator," *Entropy*, vol. 22, 2020.
- [24] I. Y. Chun and J. A. Fessler, "Convolutional analysis operator learning: acceleration and convergence," *IEEE Transactions on Image Processing*, vol. 29, pp. 2108–2122, 2020.
- [25] I. Loshchilov, "LM-CMA: an alternative to L-BFGS for large-scale black box optimization," *Evolutionary Computation*, vol. 25, no. 1, pp. 143–171, 2017.