An N & Fu W. (2025) UTILIZING DEEP LEARNING FOR THE AUTOMATED ANALYSIS OF VIDEO SURVEILLANCE IN SPORTS FORECASTING. Revista Internacional de Medicina y Ciencias de la Actividad Física y el Deporte vol. 25 (99) pp. 352-369.
DOI: https://doi.org/10.15366/rimcafd2025.99.023

ORIGINAL

UTILIZING DEEP LEARNING FOR THE AUTOMATED ANALYSIS OF VIDEO SURVEILLANCE IN SPORTS FORECASTING

Ni An[1], Wei Fu[2*]

[1] Department of Physical Education, Guangdong Pharmaceutical University, Guangzhou, Guangdong 510006, China
[2] School of physical education, Wuhan Huaxia Institute of Technology, Wuhan, Hubei 430223, China.
E-mail: Fw158136@163.com

Recibido 09 de Marzo de 2024 **Received** March 09, 2024
Aceptado 04 de Octubre de 2024 **Accepted** October 04, 2024

ABSTRACT

In order to improve the effect of sports prediction, this article applies deep learning to the automatic collection and processing of video surveillance data. The established light field multi-viewpoint real-time 3D surface reconstruction system mainly includes multi-camera joint calibration, multi-view image dense matching and depth generation. In order to improve the efficiency and accuracy of multi-view image matching, a four-direction semi-global matching cost accumulation method is proposed. After the dense matching of the multi-view images is completed, the global optimization method is used to optimize the initial depth map, remove noise and smooth the homogeneous area, and improve the quality of the depth map. The experimental research results show that the deep learning method proposed in this article has a very good application effect in the automatic collection and processing of video surveillance data in sports prediction.

KEYWORDS: deep Learning; Video Surveillance Data; Automatic Collection; Sports

1. INTRODUCTION

With the rapid development of modern competitive sports, the requirements for the comprehensive ability of athletes to create excellent results in competitions are getting higher and higher. Among them, the psychological quality in the competition is an important component of this

comprehensive ability. From the analysis of the internal factors of the competition performance, the athlete's competition performance depends on the competition performance, and the competition performance depends on the physical fitness, technical tactics and psychological state. These states are the embodiment of physical fitness, technical and tactical ability and psychological quality acquired by athletes through genetics and learning (Aso et al., 2021). Sports practice has proved that in the competition, about 70% of athletes have unsatisfactory results due to insufficient psychological preparation, only 20% of athletes have unsatisfactory results due to insufficient training level and technical preparation, and 10% of other factors. This shows that the psychological quality of the competition process plays a decisive role in the achievement of athletes. In particular, for billiards that requires sophisticated techniques, complex tactics, and one-to-one direct confrontation, higher psychological quality is required (Azhand et al., 2021). In the competition, billiards players must have good psychological qualities to be able to play their technical level normally or abnormally. Moreover, every athlete has his own psychological characteristics, and these psychological characteristics often determine his own playing style. The formation of excellent psychological qualities in competitions is accumulated in usual training and competitions. The game is ever-changing, and sometimes the outcome of the game often depends on the psychological quality. Moreover, psychological factors are often manifested in athletes' movements. Therefore, the recognition and analysis of the athletes' actions and the combination of the data collection system can collect the athletes' sports data and psychological data, which has a positive meaning for improving the athletes' competition performance and training effects (Bakshi et al., 2021). In order to improve the effect of deep sports prediction, this paper applies deep learning to the automatic collection and processing of video surveillance data to improve the processing effect of sports prediction data, thereby enhancing the effect of sports prediction.

2. Related Work

The functional movement test (FMS) and functional movement training originated from the Athletic Performance Institute in the United States. Its test purpose and function are to test and evaluate the human body’s ability to master and use basic body movements and movement structure, and to formulate and design corrective actions based on the test results The method of practice can effectively avoid sports injuries caused by wrong actions, improve the work efficiency of sports actions and the rationality of sports (Bhombe et al., 2021). Functional movement training contains 8 core contents, and "functional movement screening" is one of its core contents. Functional Movement Test (FMS) is proposed in the literature (Colyer et al., 2018) based on clinical experimental research and personal experience based on related functional sports. The results of functional action screening are used to evaluate the effect of the subjects in completing the actions, and to avoid the formation or

continuous deterioration of sports injuries to the greatest extent. The functional action screening test is composed of seven basic test actions and three additional injury inspection action tests. The scores are quantified and evaluated according to the completion of the screening actions of the subjects. Functional movement test is a technology for evaluating human functional movement. It is expected to discover the movement limitation and asymmetry of the subject's body through the completion of the screening movement by the subject (Díaz et al., 2021). This kind of evaluation technology can amplify the problem of physical compensation when the subject completes the action, and we can easily find the problem of the body. When the human body completes certain movements, the body's movement defects will lead to the occurrence of weak links in the sports system, and make the athletes inefficient to complete technical movements during competitions and training, and there is a corresponding risk of athletes' sports injuries (Ershadi-Nasab et al., 2018). FMS can be used as a test to check the athlete's body to determine the problems that may occur in the athlete's body during exercise, and that are difficult to find during traditional medical examinations. This test, which is developed based on the completion of the body's movement, can detect functional movement limitations and body asymmetry related to proprioception, balance and stability of the body. If these existing problems can be found using functional exercise screening, the athlete's body movement pattern can be changed early, thereby reducing the occurrence of athletes' sports injuries, and thereby improving the athlete's performance ability (Hua et al., 2020). Literature (Li et al., 2020) wants to confirm whether the functional action screening score is related to sports injury, so relevant researches have been carried out on athletes to record the functional action screening scores. And follow up the subjects' sports injuries throughout the season, and record the occurrence of serious sports injuries. They classify sports injuries that take more than 3 weeks of rest after injury to athletes as serious sports injuries. The results show that the results of the functional movement screening are related to the sports injuries of the subjects participating in the test. The results show that athletes with lower FMS scores are more likely to suffer sports injuries. The literature (Liu et al., 2021) carried out functional action screening, through the screening, the scores of the screening were recorded respectively to verify the relationship between the injury and the FMS score and to conduct research. The FMS score split point is set at 16 points. Literature (Liu et al., 2018) investigated and studied athletes. The subjects included football, volleyball and basketball. Before the start of the season, the subjects were screened for functional actions and the screening scores were recorded. Follow the sports team to observe the subjects' sports injuries throughout the season, and set the FMS score split point as 14 points. The above studies show that the validity of FMS is more controversial, and most studies prove that FMS can be used as a tool for predicting sports injuries. In the literature (McNally et al., 2018), in the research on functional action screening and evaluation, the FMS score segmentation point is set to 14 points.

The research results show that when the FMS screening result score is not higher than 14 points, the athletes' chances of sports injury will be Increase from 15% to 50%. For those athletes whose FMS scores are less than 14 points, it is generally recommended that special training should not be continued, and rehabilitation training to strengthen motor functions should be put in the first place, which can relieve pain and relieve pain. It can reduce the chance of athletes getting injured. Literature (Mehta et al., 2017) conducted a survey and research on professional athletes. Through a one-year follow-up study, it is proved that athletes are 2.6 times more likely to be injured in hip flexion and extension imbalance than hip flexion and extension balance. Literature (Nagalakshmi Vallabhaneni, 2021) investigated and studied athletes. The subjects included football, volleyball and basketball. Before the start of the season, the subjects were screened for functional actions and the screening scores were recorded. Follow the sports team to observe the subjects' sports injuries throughout the season.

3. Real-time 3D surface reconstruction system of light field multi-view image based on deep learning

In this paper, we combine deep learning to model the sports process. We perform algorithm improvement and spatial model construction. The established optical field multi-view real-time 3D surface reconstruction system mainly contains three parts: multi-camera joint calibration, multi-view image dense matching and depth generation, and 3D dense reconstruction visualization, and the implementation flow of the reconstruction system is shown in figure 1.

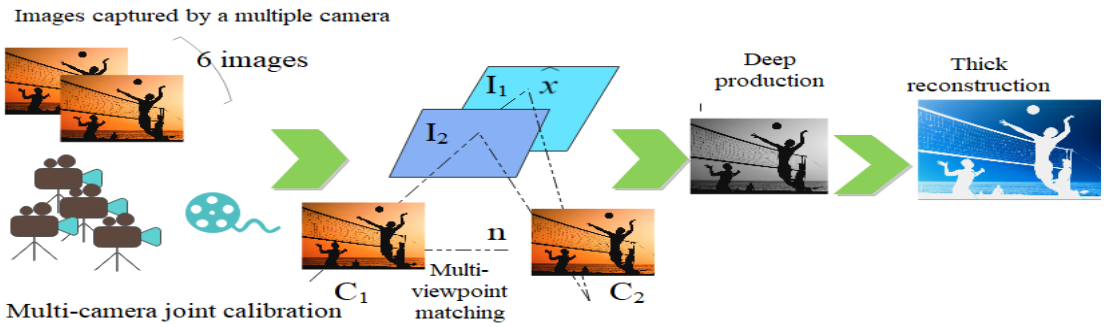


Figure 1: Schematic diagram of the realization process of the real-time 3D surface reconstruction system for light field multi-view images

Table 1 shows a partial comparison of the experimental results of the joint multi-camera calibration based on two different calibration methods, Zhengyou Zhang and VisualSFM, for the camera array shown in figure 2(b). The upper row of each parameter comparison is the parameter obtained by Zhengyou Zhang's calibration method, and the lower row is the parameter obtained by VisualSFM's self-calibration method. Comparing the results of the two multi-camera joint calibrations, we can see that (Nie et al., 2018): Although

the camera parameters obtained by the two calibration methods are approximately the same, the calibration accuracy of the joint multi-camera calibration method based on Zhang Zhengyou's plane calibration algorithm is higher, and can be seen from the inverse projection pixel error. The main reason is that Zhang Zhengyou's plane calibration algorithm makes the extraction of feature points more accurate by using a standard calibration template. This is also the main reason for adopting Zhang Zhengyou's plane calibration algorithm for the joint multi-camera calibration (Nie et al., 2019).

Table 1: Comparison of multi-camera joint calibration results based on two different calibration methods

	REFERENCE CAMERA (CAMERA 1)	CAMERA 2	CAMERA 3	CAMERA 4
INTERNAL PARAMETERS (FX, FY) (CX, CY)	490.5290 491.9580 322.0730 192.8820	490.8030 491.9950 315.1140 185.0950	490.2090 491.6790 316.4410 190.0050	493.8750 495.5150 321.8430 181.0310
	483.5020 486.3280 320 180	484.1150 484.2560 320 180	485.996 486.005 320 180	490.269492.355 320 180
TRANSLATION VECTOR (TX, TY, TZ)	000	-48.540-0.069 3.933	-96.825 -3.737- 1.991	-143.583-3.300 17.901
	000	-50.387 -0.102 3.865	-94.031 -3.882- 2.387	-146.290 -3.103 16.565
BACK PROJECTION ERROR	0.1260 0.1350	0.1200 0.1050	0.1230 0.1320	0.1280 0.1020
	0.3990 0.3640	0.4930 0.3200	0.3840 0.3660	0.5010 0.3400

The three-dimensional space of the target scene is swept by a virtual plane, and the movement direction of the plane is along the Z axis of the scene and parallel to the reference viewpoint, as shown in figure 2.

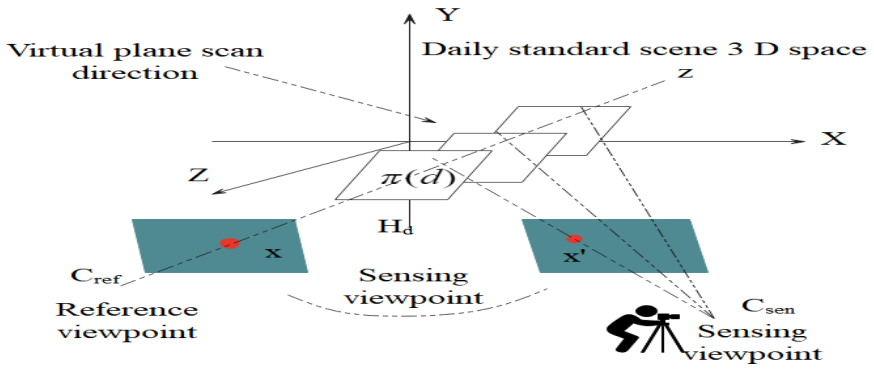


Figure 2: Schematic diagram of plane scanning algorithm for dense matching of multi-view images

In the case of constant illumination luminance. If the plane of a certain depth d happens to lie on the surface of the reconstructed object, the pixel x in the reference viewpoint image and the corresponding single-response mapped pixel x' in the sensed viewpoint image will match at this time. Therefore, the matching cost of the corresponding pixel between the reference viewpoint image and the sensed viewpoint image can be calculated to determine the depth value of each pixel in the reference viewpoint image. When the plane is scanned traversing the three-dimensional space, all pixels on the reference viewpoint image are matched and the depth values of all pixels on the reference viewpoint image can be obtained (Petrov et al., 2018). Without loss of generality, we assume that the camera internal parameter matrix corresponding to the reference viewpoint image is K_1, and the external parameter matrix is $C_1 = [R_1|t_1]$. The camera internal parameter matrix corresponding to the sensing viewpoint image is K_i, where $i = 2, \cdots 6$ and the external parameter matrix is $C_1 = [R_1|t_1]$. $n = (0,0,1)^T$ is the unit vector along the Z axis, and d is the distance (depth) relative to the reference view point image plane. From the discussed homography transformation method for a given plane, the homography matrix induced by plane $\pi(d) = (n^T, d)^T$ can be deduced as (Sárándi et al., 2020):

$$H_d = K_i(R_{rel} - t_{rel}n^T/d)K_1^{-1} \quad (1)$$

Each sensor view point image is mapped onto the plane $\pi(d)$ according to the homography matrix H_d for dense matching with the reference view point image. Figure 3 shows a schematic diagram of the non-parametric local transformation method of a 5x5 pixel block. The source Census transform needs to perform 24-pixel intensity value comparisons (figure 4(a) all pixels in the blue pixel block are compared with the center pixel in turn). However, the generalized Census transform only needs to be performed 4 times (comparison between the pixels where the arrow and the tail of the arrow are located in the figure) to achieve the same or even better results. The detailed description and performance comparison of the generalized Census transform algorithm can be found in the literature.

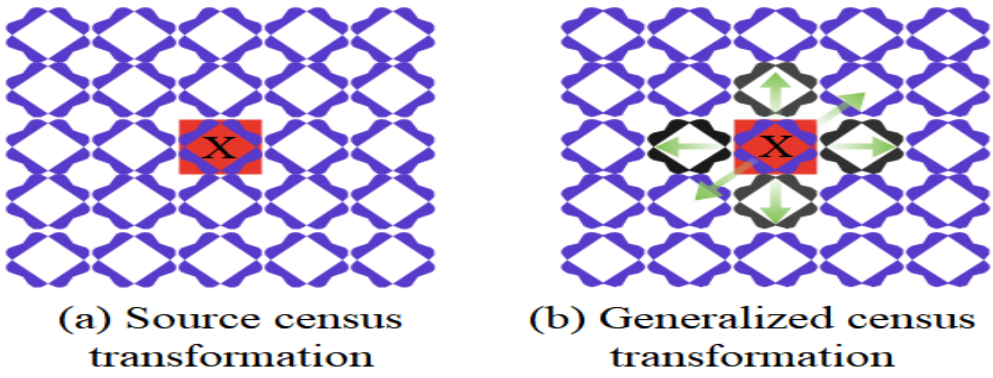


Figure 3: Non-parametric local transformation method in matching similarity calculation

The generalized Census transform of point x on the reference viewpoint image can be expressed as:

$$C(x) = \underset{1 \le j \le n}{\otimes} \xi\left(x + c_j, x + c_j'\right) \tag{2}$$

The $(c_1, c_2, \ldots, c_n)$ and $(c_1', c_2', \ldots, c_n')$ in the formula represent the defined coordinate sequence, usually n≥1. After the reference viewpoint image to be matched is subjected to the generalized Census transformation, the matching cost $cos\, t_i\,(x, d)$ is obtained by calculating the Hamming distance between the reference viewpoint image and the pixels in each sensing viewpoint image, which is expressed as (Thành & Công, 2019):

$$cos\, t_i\,(x, d) = Ham\, min\, g\left(C_1(x), C_i\left(H_i^d x\right)\right) \tag{3}$$

Formula (2) is only the matching cost between two viewpoint images. When occlusion occurs, the pixel area that is occluded is not a common area in the dual-view image, and the matching cost at this time will be meaningless. Multi-view image matching can solve the problem of occlusion in dual-view image matching. The cost of multi-view image matching is expressed as:

$$cos\, t\,(x, d) = \frac{1}{|N|}\sum_{i \in N} |cos\, t_i\,(x, d)| \tag{4}$$

It is the average value of the matching cost of multiple dual-viewpoint images, which can solve the problem of inaccurate matching when occlusion occurs. In the formula, N = 2,...,.6 is the set of sensor-viewpoint images. The cumulative path of the four-direction semi-global matching cost accumulation is up, down, left, and right. We assume that Cost, (x, d) represents the accumulation of matching costs along the path in the r direction, and r represents one of the up, down, left, and right directions. Therefore, for a certain pixel x, its initial multi-view image matching cost is known as Cost (x, d) in formula (4), and its cumulative cost $COST_r(x, d)$ on the path r can be recursively written as:

$$\begin{gathered} COST_r(x, d) = Cost(x, d) + \ min(COST_r(x - r, d - 1) + P_1 \\ COST_r(x - r, d), COST_r(x - r, d + 1) + P_1), min\, C\, OST_r(x - r, i) + P_2) \end{gathered} \tag{5}$$

In the formula, $x = (u, v)^T$ is the image pixel coordinates, and P1 is the small value penalty term, which penalizes the neighborhood pixels with small depth changes when the depth is not continuous (Xu & Tasaka, 2020). Since the cumulative cost $COST_r(x, d)$ in formula (5) increases with the addition of the minimum cumulative cost of the remaining pixels on the path, it is likely to be recursively accumulated to a huge value during the calculation process, leading to overflow. Therefore, in order to avoid overflow of the accumulated

cost value $COST_r(x,d)$ during the execution of the algorithm, the minimum path cost of the pixel x-r is subtracted from the right side of formula (5). The revised cumulative cost $COST_r(x,d)$ is expressed as:

$$\begin{aligned} COST_r(x,d) = Cost(x,d) + \ & min(COST_r(x-r,d-1)+P_1 \\ & COST_r(x-r,d), COST_r(x-r,d+1)+P_1), \\ & \min_r COST_r(x-r,i)+P_2) - \min_r COST_r(x-r,k) \end{aligned} \quad (6)$$

The cumulative cost value on the modified path r will not exceed $Cost(x,d)+P_2$, thus ensuring the smooth execution of the algorithm. In order to further improve the accuracy of matching, sub-pixel interpolation is performed in the multi-view image matching cost accumulation algorithm. The specific method is to insert sub-pixels in adjacent pixels in the same row or the same column, and the cost of performing sub-pixel matching is accumulated as:

$$\begin{aligned} COST_r(x,d) = Cost(x,d) + \ & min(COST_r(x_{sw}-r,d-1)+P_1 \\ & COST_r(x_{sw}-r,d), COST_r(x_{sw}-r,d+1)+P_1), \\ & \min_r COST_r(x_{sw}-r,i)+P_2) - \min_r COST_r(x_{sw}-r,k) \end{aligned} \quad (7)$$

In the formula, x_{sw} is the coordinate value of the sub-pixel in the defined sub-window, and the value of x_{sw} in the experiment is four values of x-0.5, x+0, x+0.5 and x+1. Finally, the four-direction sub-pixel semi-global matching cost of pixel x is accumulated as:

$$COST_{agg}(x,d) = \sum_r COST_r(x,d) \quad (8)$$

We summarize the execution steps of the above four-directional semi-global matching cost accumulation algorithm as follows: The algorithm performs a sub-pixel matching cost accumulation for all pixels in the image starting from the first pixel row $(u_i,0)$ or the first pixel $(0,v_j)$ column of the reference image toward the last row (N-1,0) or the last column (O, M-1). After that, the algorithm is repeated once in reverse (M and N are the length and width values of the reference image, respectively). Therefore, for a certain pixel to be matched, a total of four directions of matching cost accumulation are performed: up, down, left and right, as shown in figure 4. The core execution code of the algorithm in the CUDA (Compute Unified Device Architecture) parallel computing platform is shown in the Appendix (Zarkeshev & Csiszár, 2019).The initial depth value d obtained by WTA method $argmin_d\, C\, OST_{agg}(x,d)$ is often noisy and uncertain. In order to calculate a more accurate reference view depth map, we use a global method to iteratively optimize the depth generation process to remove outliers. The global energy function can be expressed as:

$$E_d = \int_\Omega \{g(x\|\nabla d\|_\varepsilon) + \lambda COST_{agg}(x,d)\}\, dx \quad (9)$$

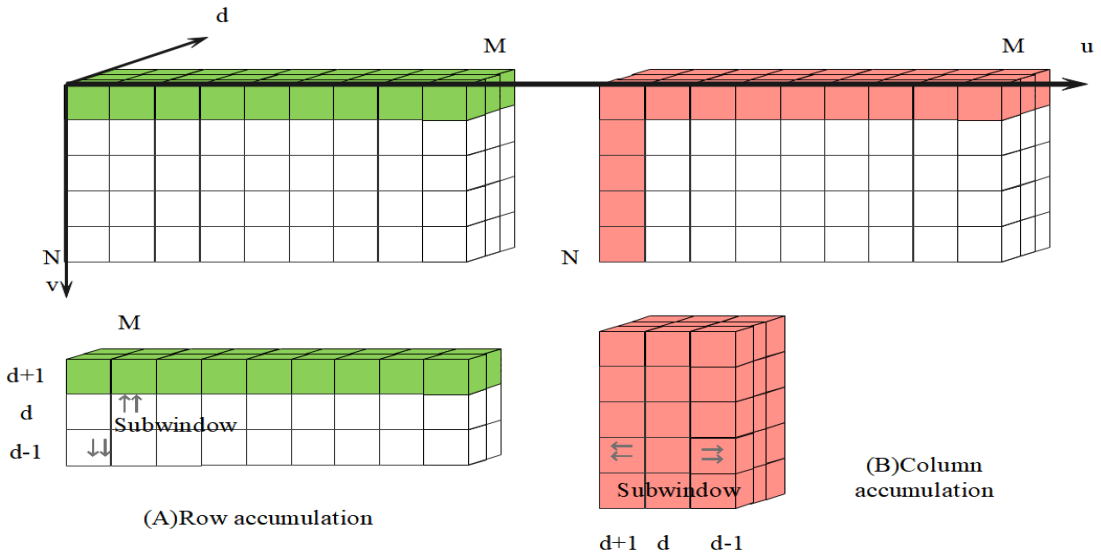


Figure 4: Sub-pixel semi-global matching cost accumulation in four directions

The data item $COST_{agg}(x,d)$ in the formula is the minimum four-direction matching cost accumulation selected by WTA. It can effectively remove noise and ensure the accuracy of the depth after denoising. The Huber norm $\|\nabla d\|_{\varepsilon}$ of the depth value gradient is used as a regular term to ensure the discontinuity of the edge while achieving a smooth and uniform area. The Huber norm is chosen because it is composed of two convex functions and has the characteristics of continuous differentiability of the quadratic norm and the robustness of the L1 norm to outliers. Since the depth discontinuity usually occurs at the edge position of the content of the reference viewpoint image, the weight of the regular term can be calculated from the gradient magnitude of the reference viewpoint image. The calculation formula is:

$$g(x) = exp(-\alpha\|\nabla I(x)\|^2) \tag{10}$$

This reduces the intensity of regularization when the edge gradient is high, and ensures that the global optimization process respects the edge. Formula (10) can be solved using the original dual algorithm, and the parallel calculation of this algorithm can ensure the real-time performance of depth generation. The detailed description of the original dual algorithm and its parallel calculation can be referred to. When solving formula (9), in order to make the regular term and data item in the equation continuous and differentiable, an auxiliary variable v is introduced to transform the equation into:

$$E_v = \int_{\Omega}\left\{g(x\|\nabla d\|_{\varepsilon}) + \frac{1}{2\theta}(d-v)^2 + \lambda COST_{agg}(x,d)\right\}dx \tag{11}$$

The θ in the formula is a very small constant. Define $Q = \frac{1}{2\theta}(u-v)^2$, which couples the original variable and the auxiliary variable together, and when

$\theta \to 0$, d=v, making the energy function formula (11) and the original energy function formula (9) equivalent.

$$\|AGd\|_{\varepsilon} = \max_{\vec{p}:|\vec{p}|_1 \le 1} \left\{ \langle AGd, \vec{p} \rangle - \delta_{\vec{p}}(\vec{p}) - \frac{\varepsilon}{2} \|\vec{p}\|_2^2 \right\} \quad (12)$$

when $|\vec{p}|_1 \le 1$. A denotes the gradient operator ∇, so that Ad computes a 2MN × 1 gradient vector. g=diag(g) is the element-by-element weighting matrix. The original dual form of the energy function is obtained by bringing formula (12) into formula (11) as follows:

$$E(d, v, \vec{p}) = \left\{ \langle AGd, \vec{p} \rangle + \frac{1}{2\theta} \|d - v\|_2^2 + \lambda COST_{agg}(v) - \delta_{\vec{p}}(\vec{p}) - \frac{\varepsilon}{2} \|\vec{p}\|_2^2 \right\} \quad (13)$$

Thus, depth generation (solving for the depth value d) eventually translates into minimizing the energy function $E(d, v, \vec{p})$. In solving for formula (13), the partial derivative of the original variable d is first performed for a fixed auxiliary variable v to obtain the condition that the stationary point d is satisfied when $E(d, v, \vec{p})$ is minimized:

$$\frac{\partial E}{\partial d} = GA^T \vec{p} + \frac{1}{\theta}(d \cdot v) = 0 \quad (14)$$

Then, to find the partial derivative of the dual variable $\vec{p}$, there are:

$$\frac{\partial E}{\partial \vec{p}} = AGd - \varepsilon \vec{p} = 0 \quad (15)$$

In the iterative process, perform the gradient descent of d and the gradient of p to obtain the updated equations of the original variable d and the dual variable $\vec{p}$:

$$\begin{cases} \vec{p}^{n+1} = \prod_{\vec{p}} \left(\left(\vec{p} + \tau_{\vec{p}} AGd^n \right) / \left(1 + \tau_{\vec{p}} \varepsilon \right) \right) (a) \\ d^{n+1} = \frac{d^n + \tau_d \left(GA^T \vec{p}^{n+1} + v^n / \theta^n \right)}{1 + \tau_d / \theta^n} (b) \end{cases} \quad (16)$$

In the formula, $\prod_{\vec{p}}(\dots)$ satisfies $\prod_{\vec{p}}(q) = q / max(1, \|q\|_2,)$, denoting the projection of the gradient ascent step onto the unit ball. θ^n satisfies $\theta^{n+1} = \theta^n(1 - 0.00h)$, τ_p and τ_d are the step lengths for the iterative update of the dyadic variable $\vec{p}$ and the original variable d, respectively. Next, the corresponding auxiliary variables $v_x (v_x = v(x) \in D)$ are found by a point-by-point search in each iteration for the primal variables d already found, such that the remaining part of the energy function is:

$$E'^{(x, d_x, v_x)} = \frac{1}{2\theta}(d_x - v_x)^2 + \lambda COST_{agg}(x, v_x) \quad (17)$$

The global optimization process of the entire depth generation can be expressed as:

1. The algorithm is initialized and iterated, and the number of iterations is n=0. We set the dual variable $\vec{p} = 0$, and initialize the original variable as:

$$d_x^0 = v_x^0 = \underset{v_x \in D}{argmin} COST_{agg}(x, v_x)$$

The algorithm starts to iterate:

2. The algorithm updates $\vec{p}$ according to the dual variable iteration formula (16a).

3. The algorithm updates d according to the original variable iteration formula (16b).

4. The algorithm uses the obtained d to calculate the auxiliary variable E through point-by-point search.

5. When θ is less than the given threshold, the algorithm iteration ends and the target depth d is output.

The core execution code of the iterative update process of the original variable d and the pairwise variable $\vec{p}$ in CUDA parallel computing is shown in the Appendix. After the depth map of the reference viewpoint image is obtained, the three-dimensional points in space corresponding to the two-dimensional image points in the depth map can be calculated through projection mapping. After that, the three-dimensional points in the space are triangulated to generate a three-dimensional patch of the scene, and texture mapping is performed on the three-dimensional patch to obtain the final three-dimensional surface model of the scene, so as to realize the visualization of the reconstruction result. If it is assumed that the two-dimensional image point coordinates in the depth map are (x, y), the corresponding spatial three-dimensional point coordinates are (X, Y, Z). According to the camera's imaging model, the projection mapping process can be expressed as:

$$\begin{cases} x = f_z\left(\frac{X}{Z}\right) + c_x \\ y = f_z\left(\frac{Y}{Z}\right) + c_y \end{cases} \tag{18}$$

The relationship between the normalized depth value d and the three-dimensional coordinate Z of the space point can be expressed as:

$$Z = \frac{1}{1/far + d \cdot step} \tag{19}$$

The step length step is obtained by step=(l/near-1/far)/LayerNum. Among them, far and near are the farthest plane value and the nearest plane value defined in the plane scan. They are set according to the real depth range

of the scene, in millimeters, and LayerNum is the number of plane scan layers. Combining formula (18) with the normalized depth value d of each pixel in the depth map, the coordinates (X, Y, Z) of all three-dimensional points in space can be calculated as:

$$\begin{cases} Z = \frac{1}{1/far+d\cdot step} \\ X = \frac{(x-c_x)\cdot Z}{f_x} = \frac{x-c_x}{f_x\cdot(1/far+d\cdot step)} \\ Y = \frac{(y-c_y)\cdot Z}{f_y} = \frac{y-c_y}{f_y\cdot(1/far+d\cdot step)} \end{cases} \quad (20)$$

Then, the three-dimensional point cloud of the scene is triangulated to obtain a three-dimensional patch. In order to reduce the complexity of the triangulation calculation and avoid confusion when selecting a triangular point cloud, the principle of the triangulation process is to triangulate each point in the depth map and the three-dimensional points corresponding to the points below and to the right, as shown in figure 5.

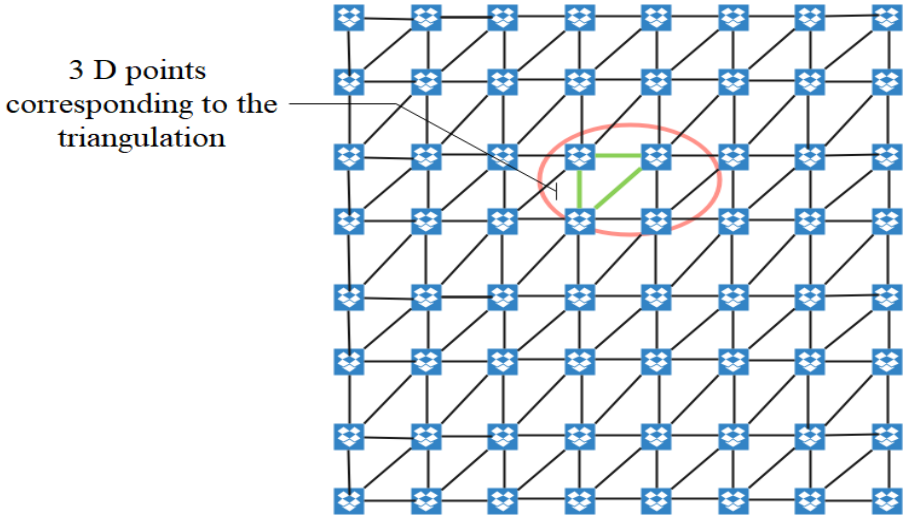


Figure 5: Triangulation of reconstructed 3D point cloud

Finally, the color value of each pixel in the reference viewpoint image is assigned as a texture to the corresponding three-dimensional point in space. Thus, the visualization of the texture of the three-dimensional surface model of the scene is realized.

4. Sports prediction system based on deep learning for automatic collection and processing of video surveillance data

This system is mainly composed of three parts: data acquisition terminal, server, and data display terminal. The system architecture is shown in Figure 1. The first part: The data collection terminal is composed of IM (Intelligent Monitor) and a camera. The IM is worn on the athlete and collects all-round monitoring

data of the athlete's health, environment, sports data and sprint prediction, and transmits it to the server. The camera sends the live signal to the server. The second part: The server stores the data information and live video data collected by IM, and sends them to the display terminal as a transfer station between the data collection terminal and the data display terminal. The third part: The data display terminal displays the data transmitted from the server in the form of webpages and APPs on large screens, mobile phones, tablets and other multimedia devices, so as to facilitate more spectators to visually watch the game and judge the results, and to have a more thorough grasp of the game status of each player.

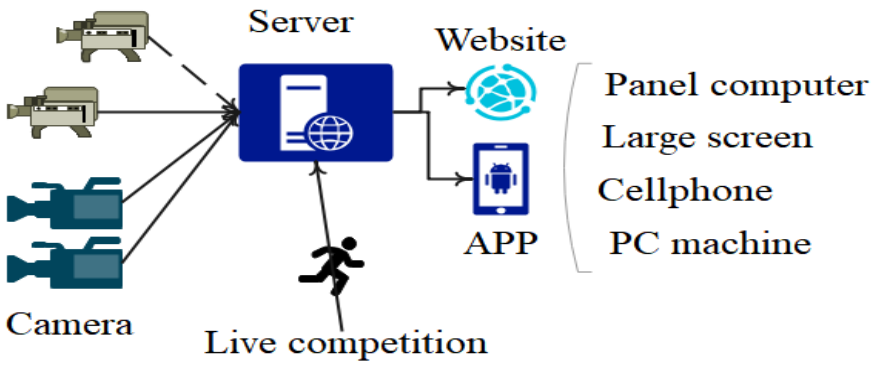


Figure 6: Block diagram of the competition system

This system selects the relational database SQLite3 as the data storage medium, which contains a total of racing athlete table, user table, competition table, and interactive platform table. The relationship between the fields and tables of each table is shown in Figure 7.

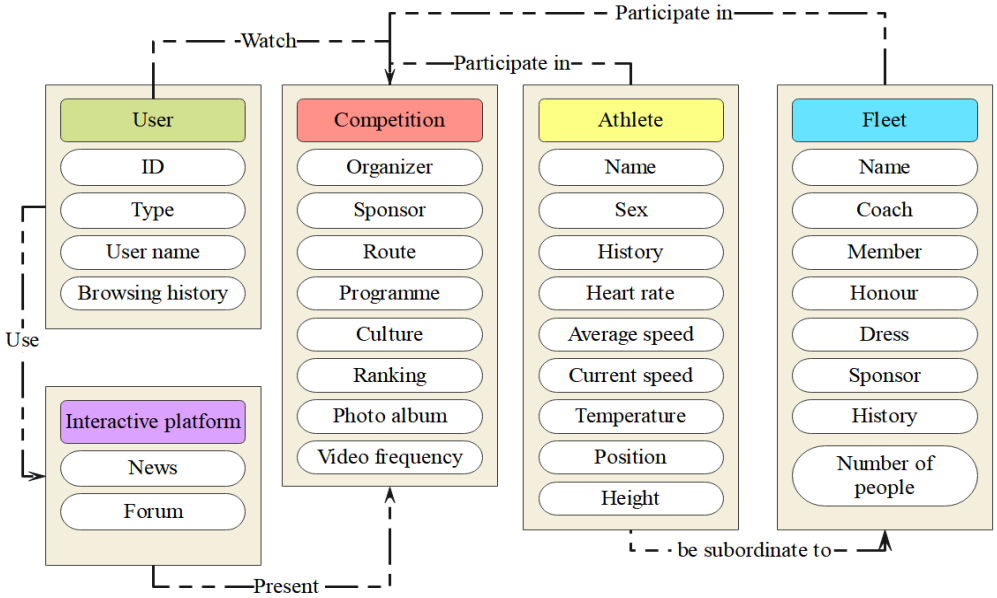


Figure 7: Database relationship diagram

On the basis of the above research, an athlete's human body model is constructed through the model in this article, and the simulation grid model is shown in Figure 8.

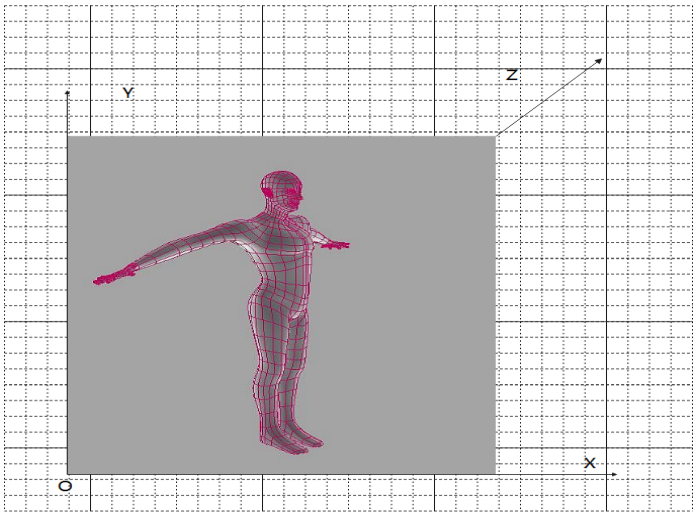
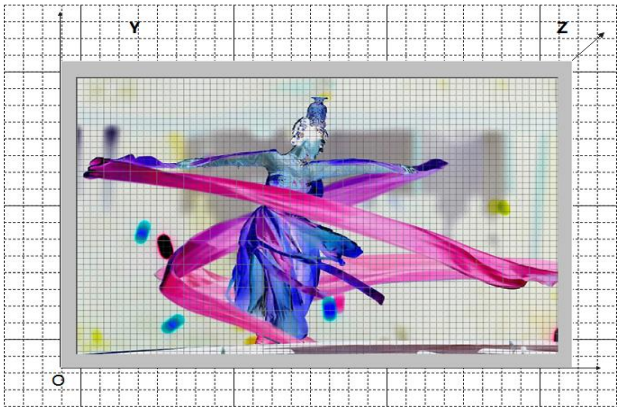


Figure 8: Athlete's human body simulation grid model

The sports data processing and monitoring simulation are carried out through the system in this paper, and the results are shown in Figure 9.

(a) Original picture



(B) The processed picture

Figure 9: Automatic collection and processing of video surveillance data based on deep learning

It can be seen from the above research that the method of automatic collection and processing of video surveillance data based on deep learning proposed in this paper has certain effects. After that, we will conduct further research on the system of this paper to explore the prediction effect of the system constructed in this paper. The statistical prediction results must be shown in Table 2 and Figure 10.

Table 2: Application effects of deep learning methods in the automatic collection and processing of video surveillance data in sports prediction

NO	FORECAST EFFECT	NO	FORECAST EFFECT	NO	FORECAST EFFECT
1	72.9431	22	77.6500	43	70.7391
2	87.7056	23	71.9429	44	75.5258
3	73.9233	24	79.6336	45	80.6946
4	77.5033	25	72.5636	46	79.8900
5	87.6829	26	88.8124	47	73.7629
6	78.4817	27	82.4861	48	73.8084
7	72.3358	28	81.9564	49	75.2886
8	78.5469	29	77.9709	50	83.2700
9	85.4432	30	79.5228	51	74.9322
10	76.5565	31	82.5291	52	74.7968
11	70.8665	32	73.6660	53	76.5001
12	71.3276	33	84.7510	54	81.1996
13	84.1116	34	90.7040	55	70.6313
14	73.3936	35	83.1911	56	81.7481
15	83.5362	36	87.1446	57	73.2668
16	76.0527	37	72.6796	58	83.6407
17	83.6818	38	71.0209	59	73.5269
18	80.3212	39	86.9717	60	72.1685
19	71.7531	40	73.3500	61	80.3252
20	81.7140	41	79.5569	62	82.8896
21	70.6680	42	72.0182	63	78.0173

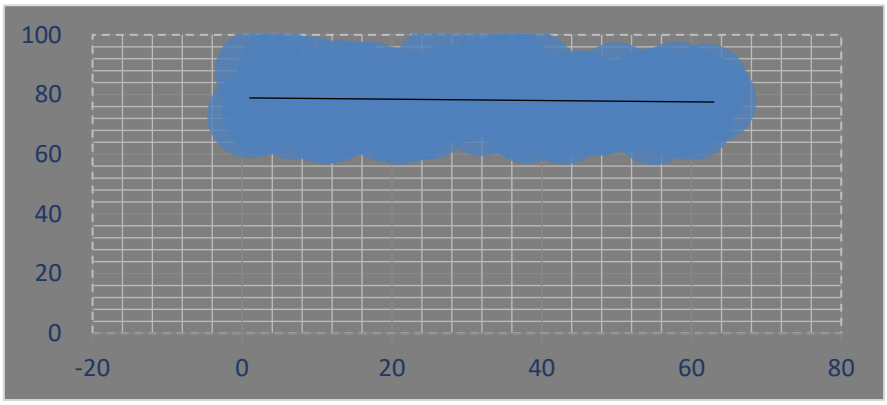


Figure 10: Simulation results of system prediction simulation

From the above simulation results, the deep learning method proposed in this article has a very good application effect in the automatic collection and processing of video surveillance data in sports prediction.

5. Conclusion

Sports competition data analysis refers to the analysis of various information displayed by the athletes, such as the skills of the athletes, based on the behavior of the athletes on the competition field. In addition to analyzing the data of the two teams and each player, it is also necessary to analyze the comparison between the players and the teams. A lot of information can be obtained in the preparation stage before the sports competition. The information that people can learn includes the previous competition results of both sides of the competition, the advantages of each athlete, and the sports skills that they have mastered, and so on.

After that, based on the recent performance of the two teams, it is predicted that the game will happen, especially the result of the game. In order to improve the effect of deep sports prediction, this paper applies deep learning to the automatic collection and processing of video surveillance data to improve the processing effect of sports prediction data, thereby enhancing the effect of sports prediction. The experimental research results show that the deep learning method proposed in this article has a very good application effect in the automatic collection and processing of video surveillance data in sports prediction

REFERENCES

Aso, K., Hwang, D.-H., & Koike, H. (2021). Portable 3D human pose estimation for human-human interaction using a chest-mounted fisheye camera. Proceedings of the Augmented Humans International Conference 2021,

Azhand, A., Rabe, S., Müller, S., Sattler, I., & Heimann-Steinert, A. (2021). Algorithm based on one monocular video delivers highly valid and reliable gait parameters. *Scientific Reports*, *11*(1), 14065.

Bakshi, A., Sheikh, D., Ansari, Y., Sharma, C., & Naik, H. (2021). Pose estimate based yoga instructor. *International Journal of Recent Advances in Multidisciplinary Topics*, *2*(2), 70-73.

Bhombe, J., Jethwa, A., Singh, A., & Nagarhalli, T. (2021). Review of pose recognition systems. *VIVA-Tech International Journal for Research and Innovation*, *1*(4), 1-8.

Colyer, S. L., Evans, M., Cosker, D. P., & Salo, A. I. (2018). A review of the evolution of vision-based motion analysis and the integration of advanced computer vision methods towards developing a markerless system. *Sports medicine-open*, *4*, 1-15.

Díaz, R. G., Laamarti, F., & El Saddik, A. (2021). DTCoach: your digital twin coach on the edge during COVID-19 and beyond. *IEEE Instrumentation & Measurement Magazine*, *24*(6), 22-28.

Ershadi-Nasab, S., Noury, E., Kasaei, S., & Sanaei, E. (2018). Multiple human 3d pose estimation from multiview images. *Multimedia Tools and Applications*, *77*, 15573-15601.

Hua, G., Li, L., & Liu, S. (2020). Multipath affinage stacked—hourglass networks for human pose estimation. *Frontiers of Computer Science*, *14*, 1-12.

Li, Z., Bao, J., Liu, T., & Jiacheng, W. (2020). Judging the normativity of PAF based on TFN and NAN. *Journal of Shanghai Jiaotong University (Science)*, *25*, 569-577.

Liu, J.-J., Newman, J., & Lee, D.-J. (2021). Using artificial intelligence to provide visual feedback for golf swing training. *Electronic Imaging*, *33*, 1-6.

Liu, S., Li, Y., & Hua, G. (2018). Human pose estimation in video via structured space learning and halfway temporal evaluation. *IEEE Transactions on Circuits and Systems for Video Technology*, *29*(7), 2029-2038.

McNally, W., Wong, A., & McPhee, J. (2018). Action recognition using deep convolutional neural networks and compressed spatio-temporal pose encodings. *Journal of Computational Vision and Imaging Systems*, *4*(1), 3-3.

Mehta, D., Sridhar, S., Sotnychenko, O., Rhodin, H., Shafiei, M., Seidel, H.-P., Xu, W., Casas, D., & Theobalt, C. (2017). Vnect: Real-time 3d human pose estimation with a single rgb camera. *Acm transactions on graphics (tog)*, *36*(4), 1-14.

Nagalakshmi Vallabhaneni, D. P. P. (2021). The analysis of the impact of yoga on healthcare and conventional strategies for human pose recognition. *Turkish Journal of Computer and Mathematics Education (TURCOMAT)*, *12*(6), 1772-1783.

Nie, X., Feng, J., Xing, J., Xiao, S., & Yan, S. (2018). Hierarchical contextual refinement networks for human pose estimation. *IEEE Transactions on Image Processing*, *28*(2), 924-936.

Nie, Y., Lee, J., Yoon, S., & Park, D. S. (2019). A multi-stage convolution machine with scaling and dilation for human pose estimation. *KSII Transactions on Internet and Information Systems (TIIS)*, *13*(6), 3182-3198.

Petrov, I., Shakhuro, V., & Konushin, A. (2018). Deep probabilistic human pose estimation. *IET Computer Vision*, *12*(5), 578-585.

Sárándi, I., Linder, T., Arras, K. O., & Leibe, B. (2020). Metrabs: metric-scale truncation-robust heatmaps for absolute 3d human pose estimation. *IEEE Transactions on Biometrics, Behavior, and Identity Science*, *3*(1), 16-30.

Thành, N. T., & Công, P. (2019). An evaluation of pose estimation in video of

traditional martial arts presentation. *Journal of Research and Development on Information and Communication Technology*, *2019*(2), 114-126.

Xu, J., & Tasaka, K. (2020). Keep your eye on the ball: detection of kicking motions in multi-view 4K soccer videos. *ITE Transactions on Media Technology and Applications*, *8*(2), 81-88.

Zarkeshev, A., & Csiszár, C. (2019). Rescue method based on V2X communication and human pose estimation. *Periodica Polytechnica Civil Engineering*, *63*(4), 1139-1146.