

Received 3 September 2023, accepted 1 October 2023, date of publication 6 October 2023, date of current version 13 October 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3322455

RESEARCH ARTICLE

Automatic Shuttlecock Motion Recognition Using Deep Learning

YONGKANG ZHAO 

Department of Physical Education, Graduate School, Woosuk University, Wanju-gun 55338, Republic of Korea

e-mail: zhaoyongkang1995@163.com

ABSTRACT In the field of sports video processing, specifically in the context of motion recognition for shuttlecock match videos, we first propose a method based on attitude estimation to measure the movement extent of the shooting arm, allowing for temporal positioning of shuttlecock shot motions and extraction of corresponding shot-sequences. The shot-sequences is defined in this paper as video segments that exclusively contain the complete shot motion of the mainplayer. During the training phase, we incorporate a lightweight channel-spatial attention mechanism into the Temporal Segment Network (TSN) to classify the extracted shot-sequences into four types, i.e., forehand shot, backhand shot, smash shot, and drop shot. Furthermore, we employ image morphology-based techniques to further distinguish smash shot-sequences as either high clear shots or kill shots based on the shuttlecock's motion trajectory. The extensive experiment results demonstrate the effectiveness of proposed method in accurately positioning and recognizing shuttlecock shot motions.

INDEX TERMS Automatic shuttlecock motion recognition, deep learning, temporal positioning, image morphology-based method.

I. INTRODUCTION

Ball sports can be classified into two categories based on the rules of winning and losing: time-based sports and score-based sports. In score-based sports, such as shuttlecock, players need to employ various techniques and footwork, including moving, jumping, rotating, and swinging their rackets on the court. The recognition of shuttlecock shot motions provides valuable information regarding players' skill levels, opponent analysis, audience experience, and data-driven training [1]. It holds significant importance for players, coaches, and spectators. Specifically:

(1) Motion analysis and technical assessment: By identifying and analyzing shot motions, players' skill levels can be evaluated. This helps coaches and players understand their strengths and areas for improvement, enabling targeted training and enhancement.

(2) Opponent strategy analysis: Recognizing shot motions allows for studying and analyzing opponents' playing styles and strategies. Understanding opponents' shot preferences,

technical characteristics, and tactical inclinations assists in formulating corresponding countermeasures and tactical arrangements, thereby improving chances of winning matches.

(3) Audience experience and entertainment value: Recognizing shot motions in shuttlecock matches enhances the visual and immersive experience for spectators. It enables better comprehension and appreciation of players' technical performances, thereby increasing the entertainment value and attractiveness of the game.

(4) Data-driven training and improvement: Collecting and analyzing a large volume of shot motion data facilitates data-driven training and improvement. Leveraging machine learning and data mining techniques, patterns and regularities can be discovered from the extensive data, providing personalized training recommendations and improvement directions for players.

In the broadcast perspective of shuttlecock matches, the trajectory of shuttlecock movement and the droppoint of shuttlecock are related to the player's shot attitude. Wang et al. [2] categorized shot motions, including high clear shots, drop shots, drive shots, and kill shots, based on the main player's


The associate editor coordinating the review of this manuscript and approving it for publication was Joewono Widjaja .



FIGURE 1. Different movements in shuttlecock match.

shot attitude in shuttlecock singles match videos. However, their classification relied solely on the gradient features of single-frame shot motion images using the Histogram of Oriented Gradients (HOG) technique, lacking motion information. Zhi et al. [3] classified the main player’s shot motions into forehand shots, backhand shots, kill shots, and other types but did not perform temporal positioning of the shot motions. In this study, we locate the shot motions of the main player (i.e., the player facing away from the camera, occupying the main region of the video) in shuttlecock singles match videos and classify them into four types: forehand shots, backhand shots, smash shots, and drop shots. Additionally, using image morphology-based techniques, we differentiate between smash shots as high clears or kills. The video sequences for each shot motion type are sampled as shown in Figure 1.

The essence of categorizing shuttlecock shot motions falls under the problem of video motion recognition in the field of artificial intelligence [4]. Video motion recognition is primarily divided into motion recognition in clipped videos and in long-duration videos [5]. Clipped videos contain only single, complete motions, while long-duration videos consist of multiple consecutive motions. In the case of long-duration video motion recognition, temporal segmentation of motions is a crucial step. These videos exhibit clear boundaries between different motions, and the foreground or background features have significant differences, as observed in behavior video

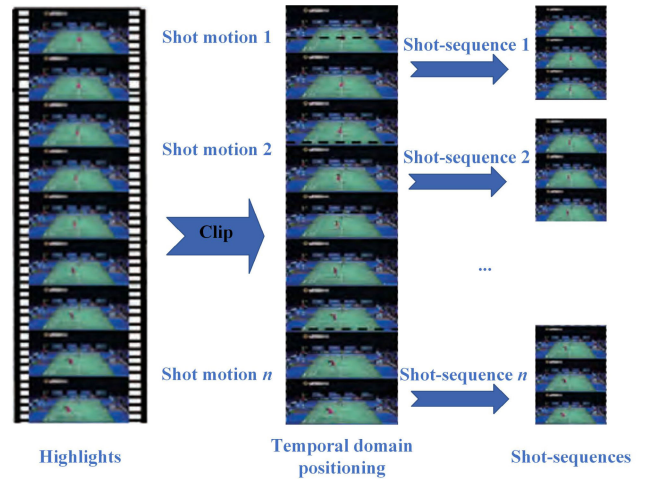


FIGURE 2. Shot-sequence processing.

Video of shuttlecock match	
Motion positioning	
Attitude estimation of arm	Quantifying the arm movement extent
Motion recognition	
Construction of shot-sequence dataset	Dual-stream extraction
Proposed CTSN model	
Result of recognition	

FIGURE 3. The overall framework.

datasets such as 50Salads [6] and Breakfast [7]. However, in shuttlecock match videos, the foreground and background features of adjacent shot motions are consistent, and there are no distinct boundaries [8]. In cases where the shot motion sequences are relatively short, the TS-WMS (time series-warp metric segmentation) algorithm [9], which measures the curvature of time series, fails to sufficiently learn the segmentation of shot motion sequences and often introduces unnecessary motion boundaries [10]. In summary, methods based on long-duration video motion recognition are not suitable for positioning shuttlecock shot motions. Therefore, a clipped video motion recognition approach is adopted for classifying shuttlecock shot motions. This paper focuses on shuttlecock match video segments and proposes a method based on multi-person attitude estimation to temporally localize and extract shot-sequences of the main player’s shot motions, as shown in Figure 2. The shot-sequence represents a video segment containing only one complete shot motion of the main player.

The overall framework of this paper is illustrated in Figure 3. For shuttlecock video segments, the player’s shot motions are localized using attitude estimation methods to identify video segments that contain a single shot motion, thus forming a shot-sequence. Then, a channel-spatial attention mechanism is introduced into the temporal segment

network, and the network is trained to classify shuttlecock motions.

In summary, our contribution is as follows:

a) By leveraging an attitude estimation model, we propose a method to temporally localize shot motions by calculating the range of arm motion, facilitating the accurate extraction of shuttlecock shot-sequences. This innovative approach does not rely on traditional image analysis or motion sensors. Instead, it utilizes arm motion range calculations for temporal positioning. This makes accurate extraction of shuttlecock shot sequences from videos more feasible and aids in subsequent analysis.

b) Through a heightened focus on feature channel information and spatial positioning information, the improved TSN with the introduction of the channel-spatial attention mechanism achieves precise classification and recognition of shuttlecock motions. This constitutes a significant contribution to video analysis in the field of sports, particularly for shuttlecock.

c) The introduced image morphology-based approach effectively discriminates between high clear shots and kill shots in the smash shot-sequences. This approach, which pays attention to the morphological characteristics of the shuttlecock in the images rather than just its trajectory, offers an effective means of distinguishing different types of shots. This method holds practical potential in the analysis of shuttlecock shot motions as it enhances shot type accuracy.

II. THE PROPOSED METHOD

A. POSITIONING OF SHOT MOTION AND EXTRMOTION OF SHOT-SEQUENCES

In shuttlecock match videos, the movement extent of the arm of the main player is larger during shot execution compared to when not shooting. Therefore, the moment of the player's shot can be located by tracking the variation in the movement extent of the arm, allowing for the extraction of shot-sequences that contain only a single complete shot motion of the main player.

To calculate the real-time movement extent of the main player's arm in a shuttlecock video sequence, accurate detection of the arm's skeletal attitude is necessary. Since there are multiple individuals in shuttlecock match videos, including players, referees, and spectators, it is not straightforward to directly locate the skeletal attitude of the main player using single-person attitude estimation algorithms. On the basis of a multi-person attitude estimation model, we add confidence, position, and joint constraints to locate the arms of the main player. First, we estimate the skeletal attitudes of all individuals in the video segment using RMPE (regional multi-person attitude estimation) [11], as shown in Figure 4. After extracting the skeletal attitudes, the arms of the main player need to be located. Among all the estimated skeletal attitudes, the two attitudes with the highest confidence belonging to the two players are selected, and then the vertical coordinate constraint is used to locate the attitude of the main player, as shown in Figure 5. Since each joint in the single-person



FIGURE 4. Extracting the skeletal attitude.



FIGURE 5. Positioning the skeletal attitude of the main player.



FIGURE 6. Estimating the arm attitude of the main player.

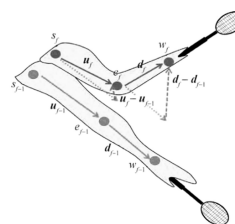


FIGURE 7. Illustration of movement vectors of the shooting arm across consecutive frames.

skeletal attitude has a fixed and unique index, the indices of the shoulder, elbow, and wrist joints are used as constraints to locate the two arms of the main player, as shown in Figure 6.

The movement extent of the arm cannot be directly obtained by calculating the Euclidean distance between arm joints in consecutive frames since it is not affected by the overall movement speed of the player. To avoid interference from changes in the player's overall position, the movement extent of the arm is calculated using the movement vectors of the upper and lower limbs of the arm.

Based on the joint features of the arm, there are three joints in the arm skeleton, namely the shoulder, elbow, and wrist, where the upper arm is between the shoulder and elbow, and the lower arm is between the elbow and wrist. In frame f ,



FIGURE 8. The detection and positioning result of the shooting arm of the main player.

the 2D coordinates of the shoulder, elbow, and wrist joints estimated by the RMPE are denoted as $s_f(x_f^s, y_f^s)$, $e_f(x_f^e, y_f^e)$, and $w_f(x_f^w, y_f^w)$, respectively. The movement vectors of the upper and lower arms are denoted as \mathbf{u}_f and \mathbf{d}_f , respectively, with vector coordinates as follows:

$$\mathbf{u}_f = e_f - s_f = [x_f^e - x_f^s, y_f^e - y_f^s] \quad (1)$$

$$\mathbf{d}_f = w_f - e_f = [x_f^w - x_f^e, y_f^w - y_f^e] \quad (2)$$

Therefore, the movement vectors of the upper arm and lower arm in frame f are $\mathbf{u}_f - \mathbf{u}_{f-1}$ and $\mathbf{d}_f - \mathbf{d}_{f-1}$, respectively, as shown by the dashed vectors pointing towards e_f and w_f in Figure 7.

The movement extent of the arm in frame f is defined as the linearly weighted sum of the squares of the norms of the movement vectors of the upper and lower arms:

$$\varphi_f = \lambda \|\mathbf{u}_f - \mathbf{u}_{f-1}\|^2 + (1 - \lambda) \|\mathbf{d}_f - \mathbf{d}_{f-1}\|^2 \quad (3)$$

in which, the parameter λ ($0 \leq \lambda \leq 1$) and $1 - \lambda$ represent the weights of the movement vectors at the elbow and wrist, respectively. The square of the norm enhances the difference in movement extent between shooting and non-shooting states.

As pointed out by [12], detecting the racket for locating the arm is impractical due to the blurry features of the racket in the video. Instead, comparing the movement extents of the two arms of the main player allows for further positioning of the shooting arm. Setting the movement extent threshold as φ_τ , the shooting arm is denoted as ζ^T , and the non-shooting arm as ζ^F . By traversing the movement extent values of each arm ζ^i ($i = T, F$) in each frame, when the movement extent φ_{f_m} in the m -th frame f_m exceeds φ_τ , f_m is marked as a movement frame for arm ζ^i . Subsequently, the frames between f_m and f_{m+t} are skipped, and the detection continues with f_{m+t} . This process continues until the traversal is completed. Here, the parameter t defines the range of frames on both sides of f_m , which determines the duration of the shot-sequence. The movement frames for arm ζ^i form an oscillation set \mathbf{F}_m^i .

If the average movement extent of the oscillation points in arm ζ^i is smaller than the other arm, it indicates that arm is generally less active than the other arm throughout the video segment, and the other arm is considered the shooting arm. Finally, the shooting arm of the main player is detected and localized, as shown in Figure 8.



(a) f_{i-1}^{RGB}



(b) f_i^{RGB}



(c) $f_i^{FLOW_X}$



(d) $f_i^{FLOW_Y}$

FIGURE 9. Example of optical flow.

To address the issue of arm occlusion in the video, RMPE can still provide reasonable estimations of the arm by referencing the visible parts of the body in frames with occlusions, as shown in the second row of Figure 8, ensuring the continuity of arm positioning in the video segment and avoiding temporal interruptions in arm detection. After positioning the shooting arm of the player, the initial and final frames of each consecutive pair of shot-sequences are referred to as the temporal boundaries of the shot-sequence. During the frame traversal process, when accessing frame f_m (s.t. $\varphi(f_m) \geq \varphi_\tau$, $\varphi(f_{m-1}) < \varphi_\tau$), since the arm motion has a certain delay, the movement extent of the subsequent adjacent frames to f_m may still exceed φ_τ . Therefore, the frames between f_m and f_{m+t} are skipped to avoid redundant movement frames. The detection continues with f_{m+t+1} until the traversal is completed. Here, the parameter t defines the frame range on both sides of f_m , which determines the duration of the shot-sequence. The movement frames of arm ζ^i form a movement set S^i . For each frame in the set S^T , when frames f_{m-t} and f_{m+t} are within the frame index range of the video segment, t frames before

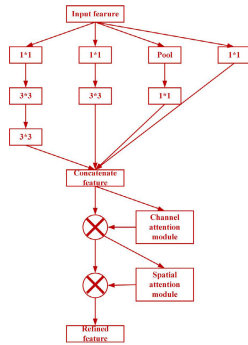


FIGURE 10. Modified module of CTSN.

and after f_m are combined to form a shot-sequence centered around f_m in the temporal domain, where frames f_{m-t} and f_{m+t} serve as the temporal boundaries of the shot-sequence. At last, the number of frames in each shot-sequence is $2t + 1$, and the value of t for the frame range can be assigned based on the frame rate.

B. SHUTTLECOCK SHOT MOTION RECOGNITION

There is a high redundancy between frames with small time intervals in the shot-sequences. If each frame of the shot-sequence is inputted into the network, it would consume excessive computational time and memory resources. The Temporal Segment Network (TSN) utilizes a random sampling strategy to reduce time and memory consumption and maintains high robustness even with limited training samples [13]. Therefore, this paper conducts research on the classification recognition of shuttlecock shot motions based on the modified TSN.

The TSN is built upon the dual-stream ConvNet (Convolutional Neural Network) [14] and requires the extraction of spatial and temporal streams from the shot-sequences after training and testing. The spatial stream contains RGB image information, which includes single-frame image information of the shooting motion [15]. The temporal stream contains global feature information of the shooting motion [16]. Subsequently, a series of small segments are obtained through random sampling, where each segment consists of one frame image and two optical flow feature maps. The obtained frame images and optical flow feature maps are separately fed into the spatial convolutional layer and the temporal convolutional layer. DenseFlow [10] is used to extract the optical flow from the shuttlecock video. The optical flow between the consecutive frames f_{i-1}^{RGB} and f_i^{RGB} is illustrated in Figure 9, where $f_i^{FLOW_X}$ and $f_i^{FLOW_Y}$ represent the horizontal and vertical components of the optical flow, respectively.

In the shot-sequence, the recognition object, i.e., the main player, occupies a partial region in each frame and exhibits spatial locality. CBAM (Convolutional Block Attention Module) [17], including spatial attention mechanism and channel attention mechanism, can enhance the generalization performance of network by focusing on important features and

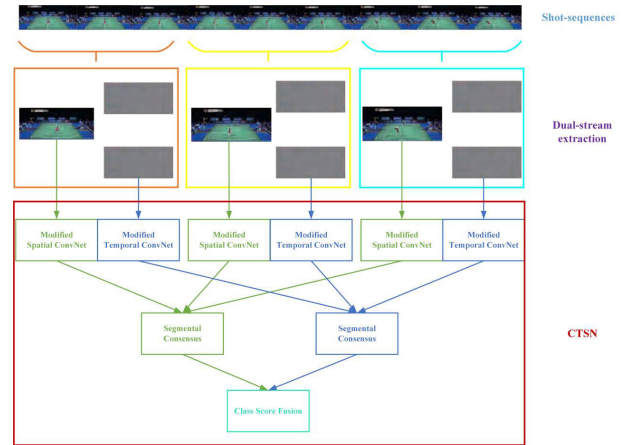


FIGURE 11. Architecture of CTSN.

suppressing irrelevant features. It can be embedded into different baseline ConvNet structures [18]. The two modules of channel attention mechanism and spatial attention mechanism can be combined in parallel or sequentially. However, previous research has shown that sequential combination with channel attention placed before spatial attention can achieve better results [19]. Therefore, this paper introduces the channel attention mechanism and spatial attention mechanism in a sequential manner. In the TSN, the spatial and temporal convolutional backbone structure used is BN (Batch Normalization)-Inception [11]. To further improve the performance of the model, Yue et al. [20] embedded SE (squeeze-and-excitation) mechanism into the Inception structure, which achieved better results. SE allows the neural network to focus on meaningful feature channels for the classification task and suppress irrelevant feature channels, but lacks focused attention on spatial positions. Inspired of this, this paper introduces CBAM (convolutional block attention module) after the BN-Inception structure, and improves the original TSN network to CTSN network, as shown in Figures 10 and 11.

Proposed CTSN architecture models the shot-sequence by segmenting it into consecutive k segments $\mathbf{T}_1, \dots, \mathbf{T}_k$. Then, a series of frame samples are randomly and sparsely sampled from each segment. Each segment provides its preliminary prediction for the motion category, and these segments are aggregated to obtain the final prediction for the entire shot-sequence. The aggregation of temporal and spatial convolutions is fused using a multi-class linear support vector machine [21].

During network training, the model parameters are iteratively updated to make the loss converge. The CTSN structure models a series of segments, as follows:

$$CTSNet(\mathbf{T}_1, \mathbf{T}_2, \dots, \mathbf{T}_k) = H(g(\mathbf{F}(\mathbf{T}_1; W), \mathbf{F}(\mathbf{T}_2; W), \dots, \mathbf{F}(\mathbf{T}_k; W))) \quad (4)$$

where H is the prediction function and W is the training parameter. The segment consensus function g aggregates the

prediction results of each segment to obtain the category prediction aggregation for the entire shot-sequence. $\mathbf{F}(\mathbf{T}_j; W)$ represents the prediction vector of segment \mathbf{T}_j ($1 \leq j \leq k$) for various types of shuttlecock shot motions. Average pooling is used as the segment consensus function [21], as follows:

$$g_i = \frac{1}{K} \sum_{k=1}^K f_i^k \quad (5)$$

where $i \in \{1, 2, \dots, C\}$, C is the total number of categories, and K is the number of segments. Based on this aggregation, f_i^k represents the prediction result of segment \mathbf{T}_k for the i -th shot motion, which is the i -th element in the vector $\mathbf{F}(\mathbf{T}_k; W)$. The prediction function H predicts the probabilities of each shuttlecock shot motion type for the entire video. Since the training set includes four types of shuttlecock shot motions, the output layer of the CTSN network consists of four nodes, corresponding to the predicted probabilities of the four types of shuttlecock shot motions. Combined with the standard classification cross-entropy loss [22], the loss function for segment aggregation is as follows:

$$L(y, G) = - \sum_{i=1}^C y_i \log H(g_i) + \frac{\eta}{2n} \sum_w W^2 \quad (6)$$

where G the prediction results of each segment that aggregating by segment consensus function g , $H(g_i)$ is the predicted probability of the model on category i , and y_i is the label value. If category i is the labeled category for a sample, y_i is set to 1; otherwise, it is set to 0. $\frac{\eta}{2n} \sum_w W^2$ is the L2 regularization term [23], which reduces the weight to a lower value for alleviating the overfitting problem of the model, n is the number of training set samples, and η is the regularization parameter. The softmax function is used for normalization in the output layer of the CTSN network [24]. The predicted category with the highest probability in the output layer corresponds to the final prediction of the CTSN network. When the shot-sequence is predicted a smash shot, it will continue to be classified as either a high clear shot or a kill shot.

High clear shots and kill shots have highly consistent posture characteristics in the game videos, as they both belong to smash shot types. However, in actual matches, they represent two distinct categories. The difference between them lies in the trend of the shuttlecock's trajectory after the shot, which is opposite to the direction of motion in the later stage. Additionally, the differences in the attitude features of the main player are evident in the shot-sequences of different shot motions. To focus the feature differences of different motion categories on the attitude features of the main player in the training set, high clear shots and kill shots are categorized as smash shots. In high clear type shot-sequences, the upper region of the frame area at the end of the video segment shows the presence of the shuttlecock's mask information. Conversely, the upper region of the frame in the shot-sequences of kill shots does not contain shuttlecock mask information.

For the image morphology-based processing method, this paper uses $\mathbf{F}_i \pm \mathbf{F}_{i-1}$ to represent the sum and difference of

pixels between frame i and frame $i-1$, respectively. $\mathbf{F} \otimes W_{n \times n}$ represents the smoothing linear filtering of frame \mathbf{F} using the operator $W_{n \times n}$. $\mathbf{F} \oplus S^r$ represents the dilation operation of image \mathbf{F} using the structuring element S^r with a radius of r . $e(w, h)$ represents an elliptical structuring element with the major axis length w and the minor axis length h . When CTSN predicts a shot-sequence as an smash shot, an image morphology-based processing method is applied to differentiate the shot-sequence as either a high clear shot or a kill shot. The specific steps for differentiating high clear shots and kill shots in the shot-sequences are as Algorithm 1.

The Algorithm 1 controls the prediction result by defining and returning the boolean variable IsHighClear. Steps 1 to 7 obtain the horizontal boundaries of the foreground court area and the upper boundary coordinate of the frame. Steps 8 to 9 determine whether the background region of each frame contains the shuttlecock mask. If a frame contains the shuttlecock mask, the ordinate of the bounding box is saved as an approximation of the mask's ordinate. Steps 10 to 11 determine whether there are at least two positive and decreasing ordinates of the mask in the frames at the end of the video. Based on this determination, the value of IsHighClear is assigned, and it is returned as the final discrimination result.

As the result, the variation of the height $h(h = y_i^R - y_i)$ between the mask region and the upper edge of the foreground court for the set of high clear shots and kill shots with respect to the frame index i is shown in Figure 12. Figure 12(a) shows the presence of consecutive mask regions with decreasing ordinates in the later part of the shot-sequence, indicating that it is a high clear shot type. Conversely, Figure 12(b) does not exhibit such consecutive mask regions, indicating that it corresponds to a kill shot type.

III. EXPERIMENT

A. EXPERIMENT SETTINGS, DATASET AND EVALUATION METRICS

Experimental environment configuration for this study included an Intel Xeon Platinum 8160Ts CPU, GeForce GTX 2080 Ti GPU, and CUDA 11 with CUDNN 7.7 as the GPU acceleration library. The deep learning framework used was Tensorflow, running on the Ubuntu system. The training parameters are listed in Table 1.

The evaluation metrics for clustering performance included Average Recall (R), Average Precision (P), and the area under the curve (AUC) for both micro-average (micro-AUC) and macro-average (macro-AUC) [25]. The formulas for calculating Average Recall and Average Precision are given by (7) and (8):

$$R = \sum_i^N \frac{TP_i}{TP_i + FN_i} \quad (7)$$

$$P = \sum_i^N \frac{TP_i}{TP_i + FP_i} \quad (8)$$

Algorithm 1 Differentiating High Clear Shots and Kill Shots in the Shot-Sequences Using Image Morphology-Based Processing Method

Input: The shot-sequence of smash shot.

Output: Predicted result.

Procedure:

Obtain the number of frames N in the shot-sequence.

Initialize a boolean control variable $IsHighClear$ as True, defaulting the discrimination result as a high clear shot.

Initialize a list \mathbf{PL} to store the ordinate of the upper-left corner of the bounding rectangle of the shuttlecock mask for each frame.

Read the first frame of the shot-sequence and segment the G channel of the frame using a threshold of 125 to obtain the channel segmentation image $\mathbf{BGR_G}$.

Remove connected regions in $\mathbf{BGR_G}$ with an area smaller than $1E+4$ and perform hole filling on the remaining connected regions to obtain $\mathbf{BGR_GF}$.

$\mathbf{FG_CL} = [\mathbf{BGR_GF} \cdot S^{50}] \otimes W_{5 \times 5}$.

Traverse the connected regions in $\mathbf{FG_CL}$ and obtain the bounding rectangle R with the largest area, then save the abscissas of left and right boundaries of R as x_l^R and x_r^R , respectively, and save the ordinate of upper boundary as y_t^R .

for $i = 1: N$:

 Read the current frame \mathbf{F}_i and its previous frame \mathbf{F}_{i-1} .

$\mathbf{F}_{i-1,i} = (\mathbf{F}_{i-1} - \mathbf{F}_i) + (\mathbf{F}_i - \mathbf{F}_{i-1})$.

 Convert $\mathbf{F}_{i-1,i}$ to a grayscale image $\mathbf{F}_{i-1,i}^{gray}$.

$\mathbf{WF}_{i-1,i}^{gray} = \mathbf{F}_{i-1,i}^{gray} \otimes W_{5 \times 5}$.

$\mathbf{WF}_{i-1,i}^{gray}$ is segmented with pixel threshold of 50, and \mathbf{WFT}_i of moving object is obtained.

$\mathbf{DF}_m^b = (\mathbf{FG_CL} + \mathbf{WFT}_i) \oplus S^{25}$.

 Define a list \mathbf{CT} to store the contours of connected regions in \mathbf{DF}_m^b .

 Traverse the connected regions in \mathbf{DF}_m^b and store each connected region in the list \mathbf{CT} .

 Remove the connected region \mathbf{B}_{\max} with the largest area from \mathbf{CT} . if \mathbf{CT} becomes empty after removing \mathbf{B}_{\max} , then $\mathbf{PL}[i] = -1$ and continue to the next iteration.

 end if.

 Define a list \mathbf{CPA} to store the local area of connected regions.

 for $j = 0:\text{length}(\mathbf{CT}) - 1$:

 Obtain the abscissas of left and right boundaries x_l^j and x_r^j , as well as the ordinate of upper boundary as y_t^j , of the bounding rectangle of connected region \mathbf{B}_j .

 if $x_l^j > x_l^R$ & $x_r^j < x_r^R$ & $y_t^j < y_t^R$:

 Store the area $S(\mathbf{B}_j)$ of \mathbf{B}_j in the list \mathbf{CPA} .

 end if.

 end for.

 if \mathbf{CPA} is not empty:

 Obtain the maximum area in \mathbf{CPA} and the corresponding bounding rectangle's ordinate y_i , and assign $\mathbf{PL}[i] = y_i$.

 else:

 Assign $\mathbf{PL}[i] = -1$.

 end if.

 end for.

 if there are at least 2 positive and decreasing elements in the last 10 elements of list \mathbf{PL} :

$IsHighClear = \text{False}$.

 end if.

 return $IsHighClear$.

where TP_i represents the i -th true positives, FN represents the i -th false negatives, FP represents the i -th false positives, and N represents the total number of motion categories.

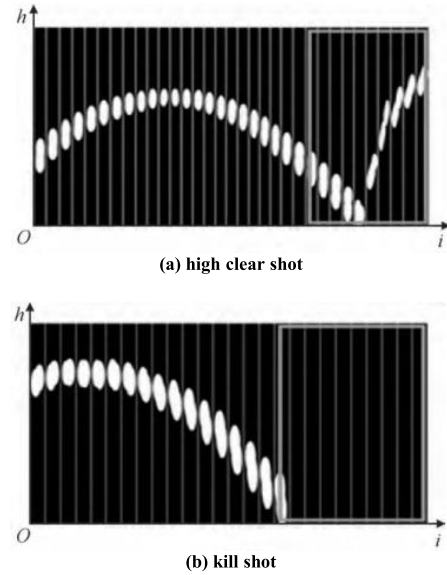


FIGURE 12. Example of height variation of shuttlecock mask in high clears and kills shot-sequences.

TABLE 1. The training parameters.

Parameters	Configurations
Epochs	50
Batch size	64
Dropout rate	0.8
Initial learning rate	$2.5 \cdot 10^{-4}$
Pretraining model	BN-inception
Optimizer	Stochastic gradient descent
L2 regularization coefficient η	$5 \cdot 10^{-4}$
Momentum	0.9

The collected shuttlecock videos were sourced from the internet, primarily including footage from the 2012 London Olympics, 2016 Rio Olympics, and the 2018 and 2019 circuit tournaments.

For the classification of motions in the shot-sequences, a manually curated dataset was created to ensure the maximum integrity of the shot motions. Using video merging and splitting software, 5,160 shot-sequences were generated from a large collection of recorded shuttlecock match videos. These shot-sequences were manually labeled with motion categories, including forehand shot, backhand shot, smash shot, and drop shot.

B. EXPERIMENT ABOUT POSITIONING OF SHOT MOTION AND EXTRACTION OF SHOT-SEQUENCES

Regarding the positioning of shot motions in video segments, the parameter λ in (3) was set to 0.35, the movement range threshold φ_τ was set to 500, and the frame range t on both sides of f_m had a value of 15. If a extracted shot-sequence \mathbf{V}_P and the ground truth shot-sequence \mathbf{V}_T

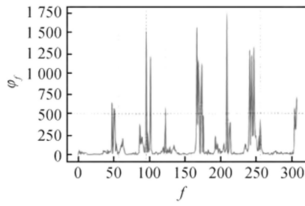


FIGURE 13. The movement extent variation curve when IoU=85.7%.

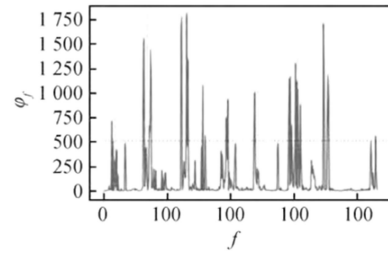


FIGURE 18. The movement extent variation curve when IoU=77.8%.

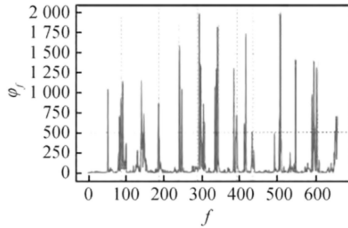


FIGURE 14. The movement extent variation curve when IoU=91.7%.

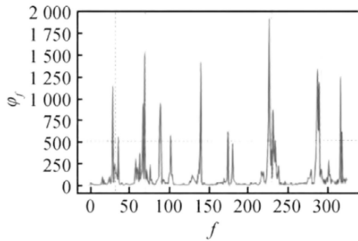


FIGURE 15. The movement extent variation curve when IoU=77.8%.

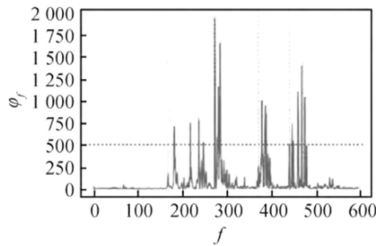


FIGURE 16. The movement extent variation curve when IoU=75%.

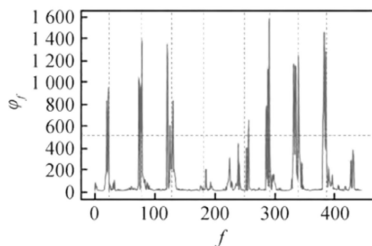


FIGURE 17. The movement extent variation curve when IoU=87.5%.

contain the same motion, then V_P is considered to be the same as V_T . For a video segment, the collection of extracted shot-sequences obtained from experimental tests is denoted as P , and the collection of true shot-sequences is denoted

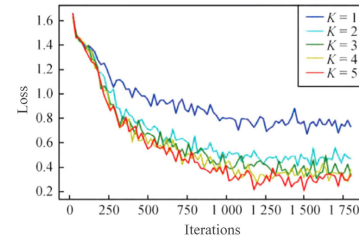


FIGURE 19. The loss change of the verification set when K takes different values.

as T . The Intersection over Union (IoU) is used to measure the overlap between sets P and T . A higher IoU indicates better performance. The movement extent variation curves of the main player’s movement arm in six video segments were measured, as shown in Figures 13-18. These frames exhibit significant fluctuations in movement extent within their neighborhoods, while the movement extents of other frames remain relatively stable.

The IoU values for the positioning of shot motions in the six video segments were measured, resulting in percentages of 85.7, 91.7, 77.8, 75.0, 87.5, and 77.8, respectively, with an average of 82.6. The IoU indicator for the positioning of shot motions in video segments is influenced by the selection of the movement extent threshold and the motion characteristics of the main player. Sometimes, when the player is not in the shot moment, the movement extent of their movement arm may still be excessive, or the movements of the elbow or wrist during the shot may not be sufficiently intense, resulting in the positioning of non-shot motions as shot motions. Such cases would result in false shot-sequences, and if they are input to the neural network for training, potentially interfering with the training and even testing. Experimental results demonstrate that the method of shot motion positioning based on movement extent discrimination shows overall good performance.

C. EXPERIMENT ABOUT SHUTTLECOCK SHOT MOTION RECOGNITION

For the testing of CTSN, we employed the hold-out method [26] and applied stratified sampling [27], which involves dividing the overall data into mutually exclusive categories and independently sampling a certain proportion of samples from each category to create a sample collection.

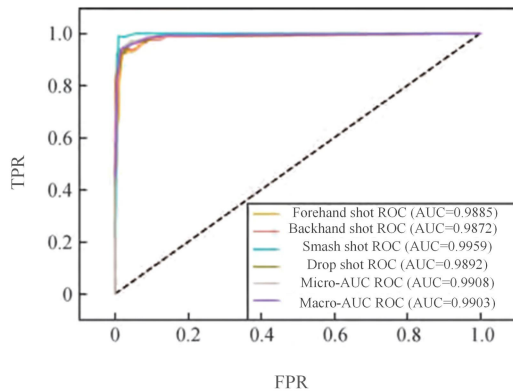


FIGURE 20. ROC-AUC.

TABLE 2. Contrast experimental results.

Methods	R/%	P/%	micro-AUC	macro-AUC
ST-GCN	87.8	89.2	0.9815	0.9826
P3D-ResNet	93.7	92.9	0.9897	0.9890
MM-SADA	93.3	93.7	0.9902	0.9910
ITN	90.1	92.4	0.9736	0.9755
SI3D	92.7	93.2	0.9689	0.9741
SA-CNN	90.7	91.5	0.9540	0.9611
CMFF	93.2	93.5	0.9900	0.9908
CTSN	93.5	94.3	0.9908	0.9903

We extracted 10% of the samples from each class of the shot-sequence dataset as the testing set, while the remaining samples were used for training. After training the network on the training set to obtain the model, we evaluated its performance using the testing set. During the training process, we used a stratified 10-fold cross-validation [28], where before each validation, the shot-sequence data from each class in the training set were divided into 10 folds, taking turns to choose one fold as the validation set and using the remaining 9 folds for network training. The learning rate was reduced to 10% of its original value at the 30-th and 40-th epochs. We set K from 1 to 5, and the training loss trends for different values of K are shown in Figure 19, with the training loss outputted every 20 iterations.

It can be observed that when K increases from 1 to 2, the convergence of the loss becomes more apparent, while when K from 2 to 5, the loss curve becomes almost stagnant, with no significant improvement in performance. At around 1250 iterations, the overall trend of the training loss stops decreasing, indicating convergence has been reached.

We find, when the number of segments K is set to 3, the recognition rate and accuracy reach a relatively balanced state. The receiver operating characteristic (ROC) curves of the classifier model for the four motion categories of forehand shot, backhand shot, smash shot, and drop shot are shown in Figure 20. The true positive rate (TPR) represents the rate of true positives, while the false positive rate (FPR) represents the rate of false positives.

TABLE 3. Confusion matrix of shuttlecock motion recognition.

True categories	Predicted categories				
	Forehand shot	Backhand shot	High clear	Kill shot	Drop shot
Forehand shot	119	1	1	0	9
Backhand shot	1	76	2	1	3
High clear	1	0	70	10	0
Kill shot	2	0	4	75	0
Drop shot	6	1	0	1	127

TABLE 4. Recall and precision of shuttlecock motion recognition.

True categories	R/%	P/%
Forehand shot	91.5	92.2
Backhand shot	91.6	97.4
High clear	86.4	90.9
Kill shot	92.6	86.2
Drop shot	94.1	91.4
Average	91.2	91.6

As shown in Figure 20, the AUC metrics for all four motion categories are above 0.98, and the micro-AUC and macro-AUC are both approximately 0.99, indicating that introducing CBAM into the TSN network and training the classifier through transfer learning can achieve good performance.

To validate the effectiveness of the proposed CTSN model in shuttlecock motion recognition, we conducted tests and comparisons with three other deep learning-based benchmark motion recognition methods. The compared methods include ST-GCN (spatial temporal graph convolutional networks) based on skeleton extrmotion [29], P3D-ResNet based on 3D convolutional networks [30], MM-SADA (multi-modal self-supervised adversarial domain adaptation) based on RGB and optical flow [31], ITN (Improved Time Network) [32], SA-CNN based on Transformer [33], SI3D (Silhouette Inflated 3D network) [34], CMFF (Context Multi-feature Fusion) [22]. To ensure fairness, we used pre-trained models and parameters suitable for each method. The experimental results are shown in Table 2.

It can be observed that P3D-ResNet achieved the highest recall rate, while MM-SADA achieved the highest macro-average, with slightly lower scores than CTSN. The CBAM-based temporal segmental network, which we proposed, achieved the highest precision and micro-average AUC. ST-GCN employs a skeleton model that is a heuristic pre-definition representing the physical structure of the human body, lacking the flexibility and capability to model multi-level semantic information contained in all layers. The experimental results demonstrate that our method, which combines the spatio-temporal features of videos with channel-spatial attention mechanisms, has certain advantages.

TABLE 5. Ablation experimental results.

Methods	R/%	P/%	micro-AUC	macro-AUC
TSN	90.2	92.5	0.9879	0.9868
SE-TSN	92.3	93.7	0.9887	0.9894
CTSN	93.5	94.3	0.9908	0.9903

To further evaluate the recognition performance of the model for each class of samples, we further divided the samples of smash shots in the testing set into high clear and kill categories. The confusion matrix of the testing set predictions is shown in Table 3, and the recall rates and precision rates for each category are shown in Table 4.

From Table 3, it can be seen that the total number of correctly predicted shot-sequence samples for each category is 467, accounting for 91.6%, demonstrating good recognition accuracy. However, there is a relatively high confusion between forehand shots and drop shots, as well as between high clears and kill shots, due to occasional similarities in the motions of forehand shots and drop shots, and when the shuttlecock at the end of high clear shot-sequences is too small or blurry, the image morphology-based method proposed in this paper is prone to misjudging high clears as kill shots. When multiple strong dynamic noises appear in the background area of the kill shot-sequence at the end, it can also lead to misjudgment of kill shots as high clears.

From Table 4, it can be seen that the recognition precision for each motion category is above 86%, and the recall rate is above 86%. The average recall rate and accuracy rate are 91.2% and 91.6%, respectively, indicating that the method based on the temporal segmental network can approach the level of human judgment to a large extent and effectively accomplish the task of recognizing shuttlecock shot motions.

D. ABLATION EXPERIMENTS

The original TSN model and the TSN models with SE and CBAM modules were subjected to ablation experiments using the testing set provided in this paper, and the results are shown in Table 5. It can be observed that CTSN achieves the relatively highest average precision and AUC, indicating that incorporating CBAM into the TSN network can improve the performance of shuttlecock motion recognition. Therefore, we adopt the proposed CTSN as the final model for shuttlecock motion recognition.

IV. CONCLUSION

We propose a method for temporal positioning and classification of shot motions performed by the main player in extracted shuttlecock video clips. The method involves detecting the player's arm using attitude estimation techniques on the shuttlecock video clips and temporally localizing the shot motions based on the variations in arm movement extent. The localized motions are used to generate shot-sequences. We introduce a channel-spatial attention mechanism into the TSN and train the network to

classify the shuttlecock motions. The classification results include four common types: forehand shot, backhand shot, smash shot, and drop shot. Additionally, we employ image morphology-based techniques to classify the smash shot as either a high clear shot or a kill shot. Experimental results demonstrate the effectiveness of the proposed method for motion recognition in shuttlecock video clips, combining temporal positioning and motion classification to enhance the intelligence of the recognition process and provide valuable applications in sports video analysis.

However, it should be noted that our recognition of shuttlecock player motions is currently limited to single-player match videos from specific broadcasting angles, which constrains the viewing perspective. Additionally, the proposed CTSN has yet to perform simultaneous classification and recognition of multiple shot-sequences. Future work will focus on developing shuttlecock video motion recognition methods that are not constrained by viewing angles and exploring parallel classification and recognition in the CTSN model to achieve real-time performance.

REFERENCES

- [1] W. Wang, "Using machine learning algorithms to recognize shuttlecock movements," *Wireless Commun. Mobile Comput.*, vol. 2021, pp. 1–13, Jun. 2021.
- [2] M. Yongkui, Z. Liang, and H. Jingxin, "Application of Kalman filter in track prediction of shuttlecock," in *Proc. IEEE Int. Conf. Robot. Biomimetics (ROBIO)*, Dec. 2009, pp. 2205–2210.
- [3] J. Zhi, D. Luo, K. Li, Y. Liu, and H. Liu, "A novel method of shuttlecock trajectory tracking and prediction for a badminton robot," *Robotica*, vol. 40, no. 6, pp. 1682–1694, Jun. 2022.
- [4] D. Y. W. Tan, H. Y. Ting, and S. B. Y. Lau, "A review on badminton motion analysis," in *Proc. Int. Conf. Robot., Autom. Sci. (ICORAS)*, Nov. 2016, pp. 1–4.
- [5] M. Ibh, S. Grasshof, D. Witzner, and P. Madeleine, "TemPose: A new skeleton-based transformer model designed for fine-grained motion recognition in badminton," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2023, pp. 5198–5207.
- [6] J. Liu and B. Liang, "An action recognition technology for badminton players using deep learning," *Mobile Inf. Syst.*, vol. 2022, pp. 1–10, May 2022.
- [7] P. Liu and J.-H. Wang, "MonoTrack: Shuttle trajectory reconstruction from monocular badminton video," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2022, pp. 3513–3522.
- [8] A. Menon, "A machine learning framework for shuttlecock tracking and player service fault detection," Nat. College Ireland, Dublin, Tech. Rep., 2023.
- [9] Y. Nokihara, R. Hachiuma, R. Hori, and H. Saito, "Future prediction of shuttlecock trajectory in badminton using player's information," *J. Imag.*, vol. 9, no. 5, p. 99, May 2023.
- [10] C.-H. Chiu, J.-L. Su, and C.-M. Lin, "The prediction of badminton flight trajectory based on an intelligent compensator," *IEEE Access*, vol. 11, pp. 32261–32271, 2023.
- [11] F. Ashfaq, N. Z. Jhanjhi, and N. A. Khan, "Badminton player's shot prediction using deep learning," in *Innovation and Technology in Sports: Proceedings of the International Conference on Innovation and Technology in Sports (ICITS), 2022, Malaysia*. Singapore: Springer Nature, 2023, pp. 233–243.
- [12] K. M. Kulkarni, R. S. Jamadagni, J. A. Paul, and S. Shenoy, "Table tennis stroke detection and recognition using ball trajectory data," 2023, *arXiv:2302.09657*.
- [13] C. Ma, D. Yu, and H. Feng, "Recognition of badminton shot action based on the improved hidden Markov model," *J. Healthcare Eng.*, vol. 2021, pp. 1–8, Oct. 2021.
- [14] W. Chen, T. Liao, Z. Li, H. Lin, H. Xue, L. Zhang, J. Guo, and Z. Cao, "Using FTOC to track shuttlecock for the badminton robot," *Neurocomputing*, vol. 334, pp. 182–196, Mar. 2019.

- [15] Y. Zhang, C. Chen, and R. Hu, "YOLO-BTM: A novel shuttlecock detection method for embedded badminton robots," in *Proc. Int. Conf. Automat., Robot. Comput. Eng. (ICARCE)*, 2022, pp. 1–6.
- [16] Y. Nokihara, R. Hachiuma, R. Hori, and H. Saito, "Future prediction of shuttlecock trajectory in badminton using Player's information," *J. Imag.*, vol. 9, no. 5, p. 99, May 2023.
- [17] J. Luo, Y. Hu, K. Davids, D. Zhang, C. Gouin, X. Li, and X. Xu, "Vision-based movement recognition reveals badminton player footwork using deep learning and binocular positioning," *Heliyon*, vol. 8, no. 8, Aug. 2022, Art. no. e10089.
- [18] J. L. Ordoñez-Avila, A. D. Pineda, J. D. Rodriguez, and A. M. Carrasco, "Design of badminton training robot with athlete detection," in *Proc. 7th Int. Conf. Control Robot. Eng. (ICCRE)*, 2022, pp. 26–31.
- [19] H. Ye, "Intelligent image processing technology for badminton robot under machine vision of Internet of Things," *Int. J. Humanoid Robot.*, Nov. 2022, Art. no. 2250018.
- [20] X. Yue, H. Li, M. Shimizu, S. Kawamura, and L. Meng, "YOLO-GD: A deep learning-based object detection algorithm for empty-dish recycling robots," *Machines*, vol. 10, no. 5, p. 294, Apr. 2022.
- [21] L. Cao and Z. Li, "Improved YOLOv5 badminton detection algorithm and embedded implementation," in *Proc. 4th Int. Conf. Robot., Intell. Control Artif. Intell.*, Dec. 2022, pp. 1316–1321.
- [22] X. Wang and J. Li, "A badminton recognition and tracking system based on context multi-feature fusion," 2023, *arXiv:2306.14492*.
- [23] H. Ma and X. Ding, "Robust automatic camera calibration in badminton court recognition," in *Proc. IEEE Asia-Pacific Conf. Image Process., Electron. Comput. (IPEC)*, Apr. 2022, pp. 893–898.
- [24] J. Zhang, "Application analysis of badminton intelligence based on knowledge graphs," in *Proc. Tobacco Regulatory Sci. (TRS)*, 2022, pp. 1004–1020.
- [25] L. Zhang and H. Dai, "Motion trajectory tracking of athletes with improved depth information-based KCF tracking method," *Multimedia Tools Appl.*, vol. 82, pp. 26481–26493, Mar. 2023.
- [26] Z. Xipeng, Z. Peng, and C. Yecheng, "Research on badminton teaching technology based on human pose estimation algorithm," *Scientific Program.*, vol. 2022, pp. 1–10, Mar. 2022.
- [27] W.-Y. Wang, T.-F. Chan, W.-C. Peng, H.-K. Yang, C.-C. Wang, and Y.-C. Fan, "How is the stroke? Inferring shot influence in badminton matches via long short-term dependencies," *ACM Trans. Intell. Syst. Technol.*, vol. 14, no. 1, pp. 1–22, Feb. 2023.
- [28] K. Mungekar, B. Marakarkandy, and S. Kelkar, "Design of an Aqua drone for automated trash collection from swimming pools using a deep learning framework," in *Proc. Congr. Intell. Syst.* Singapore: Springer Nature, 2022, pp. 555–568.
- [29] R. Hang and M. X. Li, "Spatial-temporal adaptive graph convolutional network for skeleton-based action recognition," in *Proc. Asian Conf. Comput. Vis.*, 2022, pp. 1265–1281.
- [30] B. Chen, F. Meng, H. Tang, and G. Tong, "Two-level attention module based on spurious-3D residual networks for human action recognition," *Sensors*, vol. 23, no. 3, p. 1707, Feb. 2023.
- [31] X. Liu, T. Lei, and P. Jiang, "Fine-grained egocentric action recognition with multi-modal unsupervised domain adaptation," in *Proc. IEEE 6th Inf. Technol., Netw., Electron. Autom. Control Conf. (ITNEC)*, vol. 6, Feb. 2023, pp. 84–90.
- [32] J. Zhi, Z. Sun, R. Zhang, and Z. Zhao, "Badminton video action recognition based on time network," *J. Comput. Methods Sci. Eng.*, vol. 23, no. 5, pp. 2739–2752, Oct. 2023.
- [33] Y. H. Chien and F. Yu, "Transformer on shuttlecock flying direction prediction for hit-frame detection," 2023, *arXiv:2307.16000*.
- [34] H. Zheng, H. Shen, Y. Zhang, and H. Wang, "Badminton action recognition based on improved I3D convolutional neural network," in *Proc. 3rd Int. Conf. Artif. Intell., Autom., High-Perform. Comput. (AIAHPC)*, Jul. 2023, pp. 776–784.



YONGKANG ZHAO was born in Shijiazhuang, Hebei, China, in 1995. He is currently pursuing the Ph.D. degree in physical education with Woosuk University, South Korea. His research interests include sports industry, sports economy, and sports management.

...