Low-cost Badminton Trajectory Recognition and Landing Point Prediction Based on M-YOLOv2 and Kalman Filter for Optimizing the Coordinate System Transformation of Sites

Changfen Wang
Physical Education Department, Jiangsu Vocational Institute of Commerce Jiangsu 211168, China
E-mail: wcf345@126.com

Keywords: coordinate system conversion, low-cost, badminton, trajectory recognition, landing point prediction

Received: July 20, 2024

Judging the flight trajectory curve and predicting the landing point are important factors affecting the quality of hitting in badminton matches. Many badminton professionals and coaches usually analyze the strength of badminton matches through video analysis to improve their own abilities and develop competition strategies. The study proposes a site-based coordinate transformation strategy for badminton trajectory recognition, re-examines the badminton court, and develops a method for recognizing badminton strokes by integrating the maximum residual elimination strategy with the machine learning AdaBoost algorithm. It then integrates the inherent characteristics of badminton flight to propose a badminton detection, tracking, and trajectory prediction algorithm based on video streaming. This algorithm is divided into three main components: badminton tracking, recognition detection, and trajectory prediction. The results indicated that outdoor environments had better accuracy in anchor coordinate conversion. The offset distance for anchor coordinate conversion was above 0.75m indoors and between 0.25m-0.05m outdoors. The recognition accuracy of the research method was up to more than 95% and as low as about 87% in a single environment, while the accuracy in a complex environment was up to about 92% and as low as about 80%. In terms of the running time, the duration of a single environment ranged from 1 to 2.8 seconds, while the processing time for a complex environment ranged from 2.5 to 4.5 seconds. The landing point prediction error in the proposed Kalman prediction algorithm was found to be similar to the actual landing point. The recognition accuracy was over 95%, with a minimum of around 80%, and the algorithm took between 5s to run in different environments. The improved algorithm in this study had good recognition performance with accuracy, recall, F1, and frame rate of 98.6%, 97.6%, 98.7%, and 30.8 frames/s, respectively. The research methods and results have promoting significance for the development of badminton sports.

Povzetek: Predlagana metoda temelji na algoritmih M-YOLOv2 in Kalmanovem filtru za nizkocenovno prepoznavanje trajektorij badmintonskih žogic in napovedovanje točk pristanka.

1 Introduction

Badminton is an exercise that necessitates a strategic integration of spatial, temporal, and technical elements [1]. To get a competitive advantage, badminton professionals frequently analyze game videos to acquire insights and deploy game strategies during high-level competitions. The regularity of the target's movement posture in badminton videos is weak, and existing approaches cannot accurately segment the images, resulting in low accuracy in inferring the trajectory of badminton video services [2-4]. Many scholars have put forward different opinions on this. L. Zhu proposed a badminton serve trajectory prediction based on the Fuzzy Clustering Algorithm (FCA). This method used FCA to segment badminton video images, infer the body condition ratio of moving objects and the compactness of the serving arm, construct an overall matching similarity function, and achieve tracking and trajectory prediction of moving objects. This method had good noise iteration performance and the processed motion video images had clearer targets, which could accurately predict the serve trajectory in badminton sports videos [5]. G. Cui et al. proposed a shape-based Brownian motion model and a real-time robot attitude estimation algorithm based on singular value decomposition and dynamic threshold adjustment to design badminton robots with specific weak motion targets. All algorithms could meet the real-time demands of the system, achieve simple hitting of badminton robots, and provide prospects for future research directions [6]. The advancement of artificial intelligence and the exponential expansion of the computer industry have led to a gradual increase in the practicality of Computer Vision (CV) programs. Y. Lyu and S Zhang used a CV-based badminton path tracking algorithm to analyze the Trajectory and Speed of Badminton (TSoB), aiming to use CV analysis image processing technology and path tracking. Different serving techniques had different effects on the TSoB [7].

The absence of automated data collection and suitable visualization tools resulted in a disparity between badminton coaches and athletes in the effective analysis of matches and the exchange of insights. T. Lin et al. introduced VIRD technology based on the interactive badminton game in immersive scenes, which reconstructs the game view in the game video in 3D, and proposed a top-down analysis workflow. Immersive analysis decreased context-switching expenditures and enhanced spatial acknowledge with a strong sense of presence [8]. The ability to accurately project objects at high speed was a unique skill of humans, which had become a key feature of many competitive sports. S. Vial et al. developed a model to predict the landing position of an opponent's serve by analyzing the absence of an opponent to determine whether the accuracy of object projection is affected by the opponent's presence. 69% of the opponent's serve was predicted to fall on the serve line or at close range. Therefore, the serving trajectory of elite badminton players would undergo significant changes due to the appearance of their opponents [9]. A. Sadegh et al. developed a robot that can predict the flight route of a ping-pong ball and respond properly to its return, taking into account the rotation of the ball to predict its trajectory. The qualitative results indicated that the proposed method significantly improved the prediction accuracy of ball trajectories compared to model-based methods. The prediction error of the 1st and 2nd ball landing and the estimation error of the rotation speed have been numerically confirmed to be superior [10]. Y. Gu et al. proposed a mathematical background for designing target impact trajectories when non-planar targets collide with surfaces. The established model was more accurate than other models in the case of elliptical truncated cone surfaces in oil reservoirs, and provided theoretical guidance for the design of target strike trajectories in non-planar surface oil reservoirs [11].

Traditional attitude estimation methods still suffer from imperfect occlusion, which often overlooks the continuity of human posture itself and the spatiotemporal continuity of human target trajectories. C. Cuiping considered using an improved CACShift algorithm to track athletes and proposed a video stream-based human pose estimation algorithm using badminton as an example. This algorithm estimated the pose of the human body in static images and established human tracking based on the measurement of pose distance between frames. A real-time semantic analysis scheme for badminton videos was developed using the above method, and the performance indicators of the entire system were comprehensively analyzed through experiments. It ensured tracking processing speed and reduced errors caused by tracking targets, which proved feasibility and effectiveness [12]. Accurately and reliably tracking a player's position during movements or matches was important in many sports. A. Umek and Kos A designed and carried out a real-time positioning system. It has validated UWB positioning by testing a testing application that supports tennis match strategies and analyzing the frequency of position in different areas of the tennis court during hitting. The measurement distance error between the label and two anchors showed an average of 0.1cm and 1.8cm, with variances of 14.9cm and 14.3cm. The on-site test results confirmed the practicability of UWB for position tracking in tennis and other similar sports applications [13]. A. G. Melo et al. proposed a low-cost CV algorithm to track the vertical motion of tennis balls. This algorithm aimed to determine the touch location and record the image at that moment for analysis. This method accurately detected the positions of lines and balls by applying color filters, transformations, and support vector machines, which had good performance and technical feasibility [14]. The present table tennis robot system could not decide if the ball is rotating, causing a single-return scheme for the robot and poor adaptability. H. Zhao and F. Hao proposed a table tennis target trajectory tracking algorithm that combines machine vision with a proportional conjugate gradient. This algorithm extracted 10 consecutive frames of position and velocity information for feature selection. The SCG algorithm has been improved by setting accuracy thresholds and offline learning of historical data, as well as saving the hidden layer weight matrix. Finally, the experiment verified the feasibility of the algorithm, indicating that it is more suitable for robots [15]. In other fields, trajectory tracking was also widely used. X. Yu et al. discussed the application of experimental and numerical techniques in locating free fall in a towed module model rocket with different initial landing angles. The Dropped Objects Simulator (DROBS) was utilized to calculate the ideal landing point. In both deterministic and stochastic models, the descent angle had a significant impact on trajectory, landing speed, and point. The Monte Carlo method built on random models was taken to consider the influence of various random disturbances, and the obtained distribution was analyzed through random processes for each drop angle [16]. In recent years, it has become increasingly possible to apply air transportation systems to real-world applications. Nevertheless, due to the fixed length of cables, there were practical limitations in existing papers on cable suspension transportation systems. The system with variable length urgently needed a trajectory tracking control method. H. Yu et al. designed an adaptive tracking control method that considers unknown drag coefficients. Subsequently, the convergence of the equilibrium point of the closed-loop system was proved using the Lyapunov technique and Barbalat lemma. Finally, experiments were performed on a self-established experimental platform to test the performance in air defense transportation and payload landing on mobile platforms [17].

In summary, researchers have employed a range of methods to recognize badminton trajectories and predict landing points. These include computer recognition, algorithmic classification, fuzzy clustering, singular value

decomposition, immersive analysis, predicted landing points for serves, trajectory design, machine vision, and other techniques. Additionally, the optimization of trajectory recognition accuracy has been a key area of investigation. Nevertheless, the research on the comprehensive consideration of field coordinate conversion and landing point prediction through badminton curve calculation is not yet sufficiently comprehensive. Building on existing work, the research employs artificial intelligence algorithms to analyze large data sets, introduces a new technology of field coordinate conversion, and uses virtual simulation technology to simulate experiments. Given this, experimental costs and risks are reduced and the accuracy of the results is improved. Accordingly, the study puts forth a cost-effective badminton trajectory recognition and landing point prediction optimization method founded upon the transformation of the field Coordinate System (CS). Furthermore, it introduces a novel approach to recognition research on badminton features, aiming to develop a model that will facilitate trajectory recognition.

The structure of this study is as follows: Part 1 focuses on the method and process of predicting landing points based on Field Coordinate System Transformation (FCST) and badminton trajectory recognition, which is also the focus and innovation of this study. Part 2 elaborates on the algorithm designed in Part 1, conducts experimental verification, and analyzes the data results. Part 3 summarizes the experimental results and elaborates on the lacks of this design and the directions that need to be further explored in the future.

Table 1: Summary table of related work

Reference	Author	Method	Key indicators	Result
[5]	L. Zhu	Prediction of badminton serve trajectory based on FCA	Body condition ratio, compactness of serving arm	Good noise iteration performance, clear image target, accurate prediction of serving trajectory
[6]	G. Cui et al	Real-time attitude estimation algorithm	Real time requirements and accuracy of hitting the ball	Meet real-time requirements and achieve simple hitting
[7]	Y. Lyu and S Zhang	A badminton path tracking algorithm based on CV	Trajectory and speed	Different serving techniques have varying impacts on trajectory and speed
[8]	T. Lin et al	VIRD system, 3D reconstruction of game views based on competition videos	Immersive analysis, presence, and context switching costs	Support effective game analysis, reduce switching costs
[9]	S. Vial and others	The serving model influenced by opponents	Prediction of serve landing point	The serving trajectory of excellent athletes has undergone significant changes
[10]	A. Sadegh et al	Ping Pong Trajectory Prediction Algorithm Considering Ball Rotation	Trajectory prediction accuracy and rotation speed estimation	Significantly improve prediction accuracy
[11]	Y. Gu et al	Design model for non planar target impact trajectory	Accuracy of Impact Trajectory	More accurate in the case of elliptical truncated cone surfaces
[12]	C. Cuiping	Human pose estimation algorithm based on video stream	Tracking processing speed and error rate	Ensure processing speed and reduce error rates
[13]	A. Umek and KosA	UWB positioning system, tennis match strategy testing application	Position tracking accuracy, measurement distance error	Low average error of ultra wideband system
[14]	A. G. Melo et al	Low cost CV algorithms, color filters and transformations,	Touch position detection and image analysis performance	The deployment performance of low-cost embedded computers is good

		support vector machines		
[15]	H. Zhao and F Hao	Table Tennis Target Trajectory Tracking Algorithm	Accuracy and adaptability of trajectory tracking	Improve trajectory tracking accuracy and robot adaptability
[16]	X. Yu et al	Deterministic model and Monte Carlo method based on falling object simulator	Trajectory, landing point, landing speed	Has a significant impact on trajectory and landing point, approaching the experimental distribution
[17]	H. Yu et al	Adaptive tracking control method considering unknown resistance coefficient	Trajectory tracking control accuracy, system equilibrium point convergence	The equilibrium point of the closed-loop system converges and performs well

2 Methods and materials

This study adopts the Maximum Residual Elimination Strategy (MRES) for FCST, which transforms the actual site from the model site. By analyzing the data of badminton video competitions and combining it with video streams, the algorithm for badminton detection, tracking, and trajectory prediction is studied. The analysis mainly includes badminton tracking, recognition and detection, and trajectory prediction.

2.1 FCST Based on MRES

This study simulates a badminton court and sets an anchor point as the control point for coordinate transformation based on video frame images obtained from badminton game videos, with the court center as the beginning

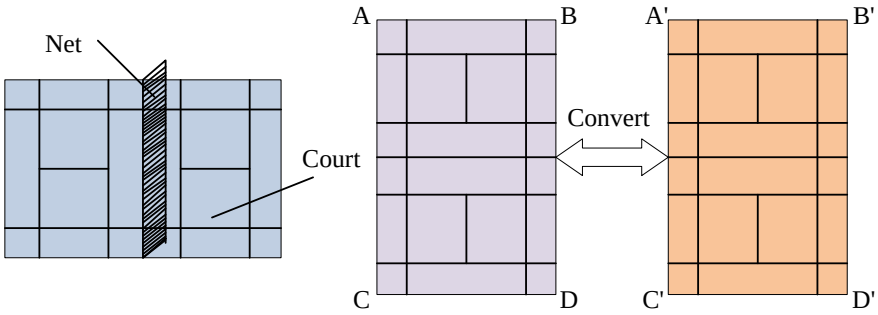


Figure 1: Badminton court model diagram

of the system. The field in badminton video shooting appears to be a flat surface, while the CS of the field is in another 2D CS, thus achieving the conversion between the image CS and the field CS. Fig.1 shows the stadium model and CS transformation.

Taking the anchor point as the control point of coordinate conversion, the conversion parameters between the image CS and the ground CS can be calculated. The road surface within the video shooting range is approximated as a plane, then the conversion of the image CS and ground CS is actually a conversion between two 2D CSs in different planes. Therefore, a single reactive transformation can be used to solve the coordinate transformation matrix. In Fig.1, Homography Transformation (HT) is simply understood as describing the positional mapping correlation of an object between the world CS and the pixel CS, and the corresponding transformation matrix is called the homography matrix [18].

This study sets the anchor site coordinates as (x, y), corresponding to homogeneous coordinates (x_1', x_2', x_3'), and the anchor image coordinates, corresponding to homogeneous coordinates (x_1, x_2, x_3). This study uses HT to calculate coordinate transformation, and the transformation relationship between anchor coordinates is given by equation (1).

$$\begin{pmatrix} x_1' \\ x_2' \\ x_3' \end{pmatrix} = H \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} \quad (1)$$

In equation (1), H is a coordinate transformation matrix of 3×3. Then, the homogeneous coordinates corresponding to the image coordinates are substituted into equation (1) to obtain the 2D site coordinates, expressed as equation (2).

$$\begin{cases} x = \dfrac{x_1'}{x_3'} = \dfrac{\dfrac{h_{11}}{h_{33}}u + \dfrac{h_{12}}{h_{33}}v + \dfrac{h_{13}}{h_{33}}}{\dfrac{h_{31}}{h_{33}}u + \dfrac{h_{32}}{h_{33}}v + 1} \\ y = \dfrac{x_2'}{x_3'} = \dfrac{\dfrac{h_{21}}{h_{33}}u + \dfrac{h_{22}}{h_{33}}v + \dfrac{h_{23}}{h_{33}}}{\dfrac{h_{31}}{h_{33}}u + \dfrac{h_{32}}{h_{33}}v + 1} \end{cases} \quad and \quad h_{33} \neq 0 \quad (2)$$

Equation (2) is further transformed into equation (3).

$$\begin{pmatrix} x \\ y \end{pmatrix} = \frac{\begin{pmatrix} a & b \\ c & d \end{pmatrix}\begin{pmatrix} u \\ v \end{pmatrix} + \begin{pmatrix} e \\ f \end{pmatrix}}{1 + \begin{pmatrix} m & n \end{pmatrix}\begin{pmatrix} u \\ v \end{pmatrix}} \quad (3)$$

In equation (3), a, b, c, d, e, and f are coordinate transformation parameters. Theoretically, for 8 unknown parameters, 4 control points can be selected to solve the coordinate transformation matrix. In practical application, the number of effective anchor points can be set to be more than 4 (e.g., this paper is set to be not less than 7). The Least Squares Method (LSM) can be used to solve the problem, which can enhance the accuracy and reliability of the calculation results. This study uses LSM to optimize the accuracy of the calculation results and enhance reliability. Inevitably, the detected coordinate values of the anchor point images will carry certain random errors, and may even be mixed with roughness. Therefore, before calculating the coordinate transformation parameters of each frame, it is necessary to formulate a suitable anchor selection strategy. This entails screening and filtering the anchor points involved in the calculation, and excluding those that exhibit roughness or significant random errors. Fig.2 shows the coordinate transformation based on MRES.

In Fig.2, first read a frame of image as raw data to obtain the corresponding anchor image coordinates and site coordinates. Determine if the requirements are met: If not, assign the previous frame result directly; If satisfied, perform coordinate transformation parameter calculation and find the optimal residual. Determine if the maximum residual is less than the set threshold: If so, save the obtained coordinate transformation parameters and proceed to the next frame of the image, repeating the operation; If not, eliminate the anchor point corresponding to the maximum residual and return a judgment on whether it meets the requirements. Repeat the operation until all frames are completed.

After calculating the coordinate transformation parameters for each frame of

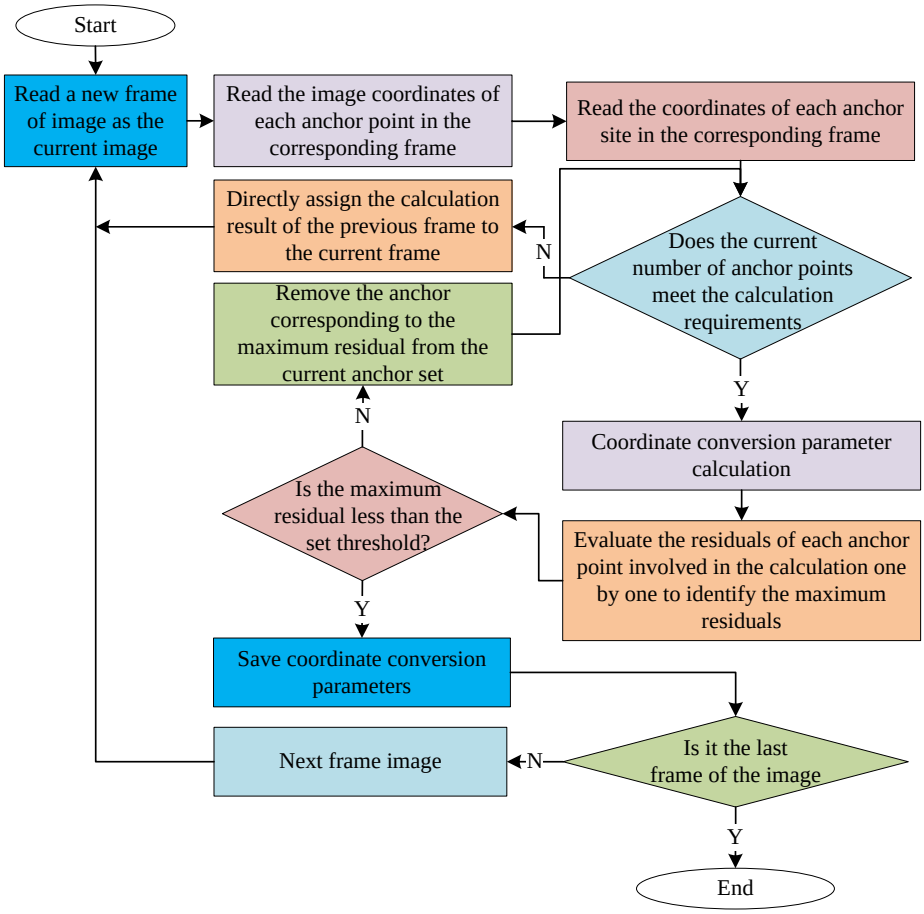


Figure 2: Coordinate transformation parameter calculation framework based on MRES

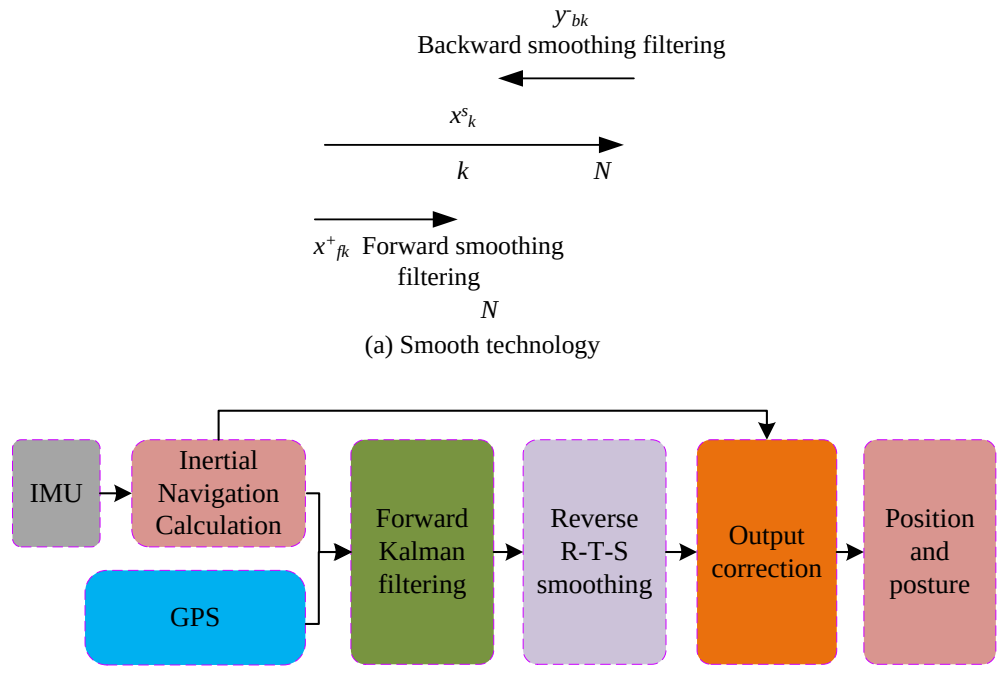
the image, the corresponding field CS can be mapped to obtain the trajectory of the badminton. However, the curve may be disturbed, so this study uses smoothing techniques to reduce errors [19]. As a method of post-processing or quasi real time data processing, smoothing technology can enhance the accuracy of data processing and has been widely used in the field of surveying and mapping [20]. The most widely used method in data post-processing in smoothing techniques is fixed interval smoothing, as shown in Fig.3. The Rauch–Tung–Striebel smoother (R-T-S) algorithm is a fixed interval optimal smoothing algorithm. The accuracy of its fixed interval smoothing algorithm outperforms Kalman Filtering (KF), and it is computationally simple and easy to do, making it an effective post-processing method [21-22]. Fig.3 shows the principle of smoothing technique and R-T-S smoothing technique. The R-T-S algorithm contains filtering of forward and backward. Forward filtering is a classic KF utilized to assess the status at each moment. Backward filtering is the reuse of partial data grounded on the forward filtering to get more precise state assessment values [23]. R-T-S smoothing can be used to calculate the following closed form solutions, as shown in equation (4).

$$p(x_k \mid y_1 : T) = N(x_k \mid m_k^s, P_k^s) \tag{4}$$

Using observation $S_y\_\{1:T\}$ to estimate the posterior distribution of the system state at time k, this process is separated into two steps: forward and backward. The Forward Recursion (FR) is given by equation (5).

$$\begin{cases} m_k' = A_k m_{k-1} \\ P_k' = A_k P_{k-1} A_k^T + Q_k \\ S_k = (H_k P_k' H_k^T + R_k)^{-1} \\ K_k = P_k' H_k^T S_k \\ m_k = m_k' + K_k (zk - H_k m_k') \\ P_k = (I - K_k H_k) P_k' \end{cases} \tag{5}$$

The Backward Recursion (BR) is given



(a) Smooth technology

(b) R-T-S smoothing

Figure 3: Smoothing technology and R-T-S smoothing

by equation (6).

$$\begin{cases} m_{k+1} = A_k m_k \\ P_{k+1} = A_k P_k A_k^T + Q_k \\ G_k = P_k A_k^T (P_{k+1})^{-1} \\ m_k^s = m_k + G_k (m_{k+1}^s - m_{k+1}^-) \\ P_k^s = P_k + G_k (P_{k+1}^s - P_{k+1}) G_k^T \end{cases} \tag{6}$$

After completing T FR from initial time 1 to time T, and then T BR from time T, the R-T-S smoothing process is achieved. Among them, the FR process is the KF process. The state estimate $\{m\}\_\{T\}$ and covariance matrix $\{P\}\_\{T\}$ obtained by FR at the last

T moments are the initial state estimate $\{m\}\_\{T\}^{s}$ and covariance matrix $\{P\}\_\{T\}^{s}$ of the BR process, i.e. $\{m\}\_\{T\}=\{m\}\_\{T\}^{s}$, $\{P\}\_\{T\}=\{P\}\_\{T\}^{s}$ [24].

2.2 Data analysis-based badminton tracking and landing point prediction algorithm

Based on the above transformation of the implementation site and model site, this study proposes a badminton detection, tracking, and trajectory prediction algorithm based on video image data. This chapter is mainly divided into several parts: badminton tracking, recognition and detection, and trajectory prediction. Firstly, badminton tracking is conducted. In this study, image differencing method is used for foreground extraction, as shown in Fig.4.

In Fig.4, the image differencing algorithm is first used for background modeling, which is based on adaptive Gaussian background modeling techniques. After obtaining the image difference, the current frame image difference map is binarized and the corresponding foreground and background pixels are determined. Then, filtering and other operations are performed on the foreground part to remove noise, and finally a clear foreground image is obtained. To extract image features, this study adopts the Adaboost algorithm, which is based on a Haar image feature description and divided into center, edge, linear, and symmetrical line features [25-26]. This algorithm is used to extract image features and obtain Haar integral maps to reduce feature computation time.

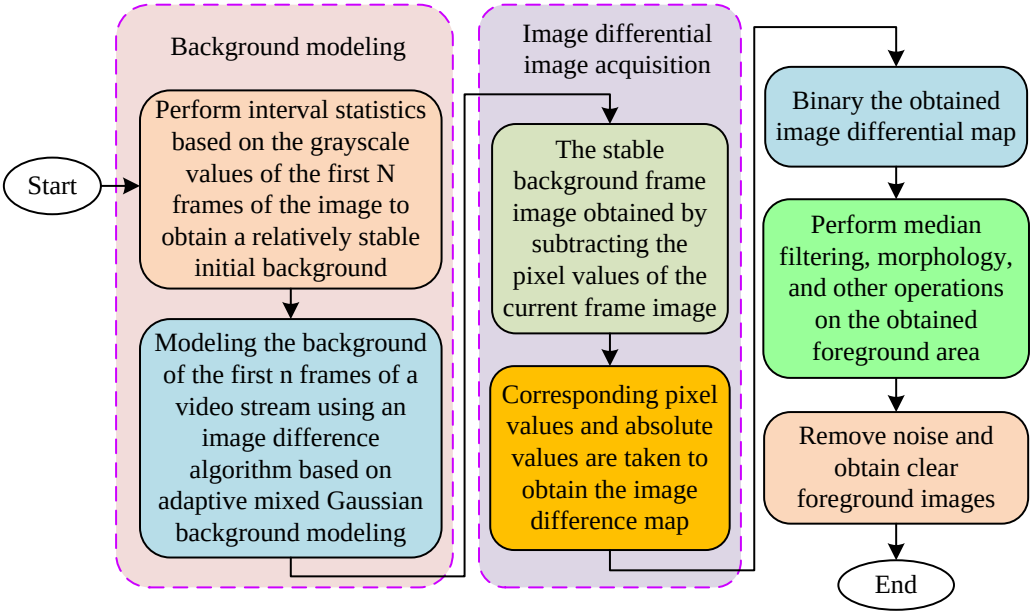


Figure 4: Image difference method for foreground extraction

The calculation at any position in the image is given by equation (7).

$$P(x,y)=P(x-1,y)+P(x,y-1)-P(x-1,y-1)+f(x,y) \quad (7)$$

In equation (7), $f(x,y)$ and $P(x,y)$ are the pixel values of the original image (x,y) and the integrated image point (x,y). The cumulative sum of pixel values in a rectangular area of a certain part of the Haar integral map is calculated using equation (8).

$$S(ABCD)=S(C)+S(A)-S(B)-S(D) \quad (8)$$

In equation (8), A, B, C, and D are the four corners of a rectangle. Therefore, a Haar integral graph is used to describe the image features of badminton as the training set for AdaBoost, and a badminton recognition model is constructed [27]. However, badminton has problems such as complex background, rapid movement, and small targets during the movement, which makes it difficult to recognize the trajectory of badminton. This study designs a fast object center tracking algorithm based on AdaBoost algorithm and traditional three frame difference, combined with Euclidean distance analysis of badminton perimeter area and other features. AdaBoost calculates through cascaded classifiers, and the repeated process for i-th iteration is as follows: the first is to normalize the input training sample weights, and the

calculation process is given by equation (9).

$$w_{i,m} = \frac{w_{i,m}}{\sum_{m=1}^{n} w_{i,m}} \tag{9}$$

In equation (9), N and n are the total number of iterations and samples. $w_{i,m}$ is the initial weight of the i -th and m -th training data. The calculation for obtaining the best weak classifier with the minimum error rate is given by equation (10).

$$\begin{cases} \varepsilon_i = \sum_{m=1}^{n} w_{i,m} \mid h_i(x_m) - y_m \mid \\ h_i = \arg\min_{h_i(x_m)} (\varepsilon_1, \varepsilon_2, ..., \varepsilon_n) \end{cases} \tag{10}$$

In equation (10), ε_i is the feature weighted error rate. h is a weak classifier. h_i is the best weak classifier. Then, the overall sample weight formula is updated as shown in equation (11).

$$\begin{cases} w_{i+1,m} = w_{i,m} \delta_i^{1-e_m} \\ \delta_i = \dfrac{\varepsilon_i}{1-\varepsilon_i} \end{cases} \tag{11}$$

Finally, a cascaded classifier is generated to construct a badminton feature model, as shown in equation (12).

$$\begin{cases} H(x) = \begin{cases} 1 & if \sum_{i=1}^{N} \alpha_i h_i(x) \geq \sum_{i=1}^{N} \alpha_i \\ 0 & otherwise \end{cases} \\ \alpha_i = \log \dfrac{1}{\delta_i} \end{cases} \tag{12}$$

An improved You Only Look Once (YOLO) algorithm has been designed for the detection of badminton, to detect the speed and accuracy of the badminton and obtain accurate coordinates. Compared to YOLOv8, YOLOv2 uses high-resolution classifiers in video stream recognition, training first on smaller resolution images and gradually increasing to the resolution required by the object detection model. This process helps the network better adapt to high-resolution inputs, thus YOLOv2 has more accurate detection performance. Due to the YOLOv2 running faster and simpler than YOLOv8, this study improves YOLOv2, mainly in terms of loss function and network structure. Due to the YOLOv2's error localization, this study optimizes the output of YOLOv2 by improving the sum of squared errors in the loss function. To address the issue of the flying badminton balls in the video stream being small targets, the YOLO simplified version of the network has been enhanced in terms of both the loss function and

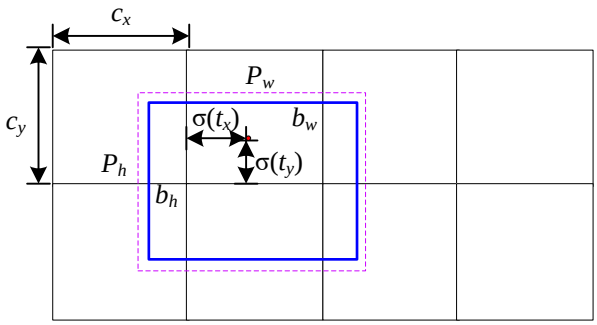


Figure 5: Schematic diagram of YOLOv2 calculating the position of boundary candidate prior boxes

the network structure. This has been done with the aim of improving the badminton detection performance, in terms of both speed and accuracy, in order to obtain more valid badminton coordinates.

The resolution of badminton is around 30×30 pixels, so this study uses YOLOv2's predicted bounding box to calculate the loss function. In addition, to improve detection efficiency, the classification part of the loss function has been removed, as shown in equation (13).

$$\begin{aligned} L = {} & \lambda_{coord} \sum_{i=0}^{S^2} \sum_{j=0}^{B} 1_{ij}^{obj} \left[\left(x_i - \hat{x}_i \right)^2 + \left(y_i - \hat{y}_i \right)^2 \right] + \\ & \lambda_{coord} \sum_{i=0}^{S^2} \sum_{j=0}^{B} 1_{ij}^{obj} \left[\left(w_i - \hat{w}_i \right)^2 + \left(h_i - \hat{h}_i \right)^2 \right] + \\ & \sum_{i=0}^{S^2} \sum_{j=0}^{B} 1_{ij}^{obj} \left(C_i - \hat{C}_i \right)^2 + \lambda_{noobj} \sum_{i=0}^{S^2} \sum_{j=0}^{B} 1_{ij}^{noobj} \left(C_i - \hat{C}_i \right)^2 \end{aligned} \tag{13}$$

In equation (13), 1_i^{obj} is whether the object appears in cell i. 1_{ij}^{noobj} and 1_{ij}^{obj} are used to detect and predict the absence and presence of objects in the j -th bounding box of unit i. $p_i(c)$ is the proportion of objects with a probability of c appearing in cell i. C is the confidence score. (w, h) and (x, y) are the width/height and center coordinate of the bounding box. λ_{coord} and λ_{noobj} are introduced to enhance network stability. The value with the symbol $\wedge$ represents the true value, while the value without a symbol $\wedge$ is the predicted value [28-29].

Therefore, this study attempts to enhance the M-YOLOv2 to obtain a new detection network structure, as shown in Table 2. This is because when improving the network structure of YOLOv2, excessive use of pooling layers can weaken the semantic features of badminton images obtained through network structure learning.

Among them, "CL" represents the convolutional layer, and "MpL" represents the max pooling layer.

In Table 2, the final CL uses a traditional activation function, with one MpL added every four layers in the previous CLs. The final stage of the detection process uses the Softmax

Table 2: The network architecture for enhancing M-YOLOv2 algorithm

Layer Type	Output	Filters	Layer Type	Output	Filters
Input	416×416	-	MpL	26×26	-
CL	416×416	4	CL	26×26	64
CL	416×416	4	CL	26×26	64
CL	416×416	8	CL	26×26	128
CL	416×416	8	CL	26×26	128
MpL	104×104	-	MpL	13×13	-
CL	104×104	16	CL	13×13	256
CL	104×104	16	CL	13×13	256
CL	104×104	32	CL	13×13	30
CL	104×104	32	/	/	/

function to generate the output. In the trajectory and landing point prediction algorithm of badminton, due to the influence of gravity, resistance, etc. during the movement of badminton, constructing the flight curve of badminton is a relatively complex nonlinear model [30-31]. This study uses LSM to achieve badminton trajectory prediction, and then optimizes the badminton prediction points through KF. LSM is a mathematical tool widely utilized in various fields of data processing like error estimation, uncertainty, system identification and prediction, and forecasting [32-33]. The specific steps of using LSM in this study are as follows: first, a series of data needs to be set to obtain the optimal function, and then the sum of squared errors between data points is fitted, summarized as equation (14).

$$\arg\min_{\vec{\beta}}\left(\vec{\beta}\right)=\left\|f(\vec{x};\vec{\beta})-\vec{y}\right\|_2^2 \tag{14}$$

Among them,

$$\vec{x}=\begin{pmatrix}x_1\\x_2\\\cdots\\x_n\end{pmatrix},\vec{y}=\begin{pmatrix}y_1\\y_2\\\cdots\\y_n\end{pmatrix},\vec{\beta}=\begin{pmatrix}\beta_1\\\beta_2\\\cdots\\\beta_n\end{pmatrix} \tag{15}$$

In equations (14) - (15), β is the coefficient vector, and $\vec{\beta}$ represents the optimal coefficient vector. $\vec{x}$ and $\vec{y}$ are vectors of the x-coordinate and y-coordinate of the badminton point. Then, LSM is used for trajectory fitting to solve the trajectory curve and predict the landing point. Due to the insufficient accuracy of existing models, this study uses KF for optimization to achieve more accurate landing point prediction. The schematic diagram of KF is displayed in Fig.6.

In Fig.6, KF is an algorithm that utilizes the state equation of a linear system to estimate the optimal state of the system through input-output observation data and feedback control. Due to the presence of noise and interference in the observed data, the optimal estimation can also be seen as

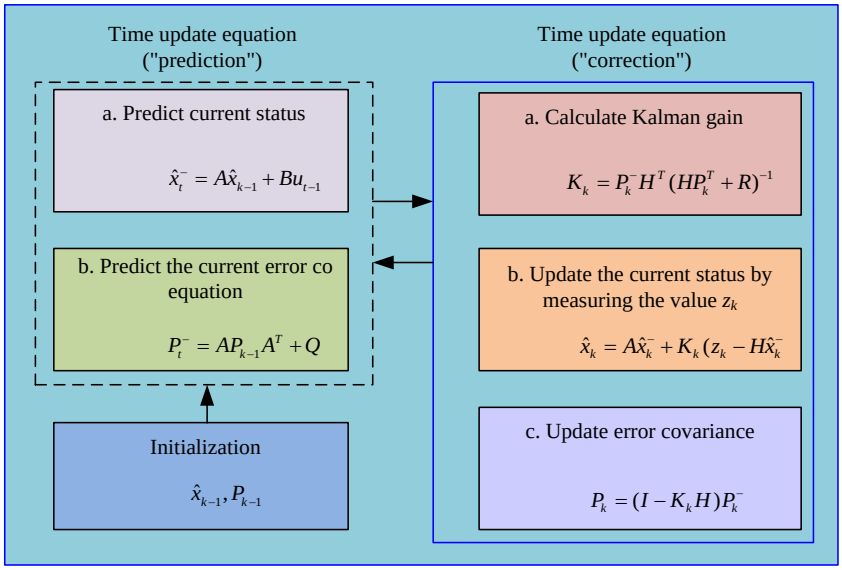


Figure 6: Kalman filtering block diagram

a filtering processes. The time update equation is used for prediction and correction, mainly to predict the current system state and error co equation [34]. Correction first calculates the Kalman gain, and updates the current state and error co equation through measurement values.

3 Results

To verify the proposed low-cost badminton trajectory recognition and landing point prediction optimization method based on FCST, an experiment was conducted to validate it. The experiment analyzed the corresponding design parameters and experimental data results, verified the advantages and feasibility of the method, and provided reference for designers to design product shapes.

3.1 Data source and processing of badminton trajectory recognition video

The experiment trained badminton video training samples to obtain a badminton feature model with 100 iterations. To obtain the optimal set of weak classifiers and then combine them to form a cascade classifier, this study adopted an evaluation mechanism at the end of each iteration to automatically select the weak classifier with the best performance in the current round as the "best weak classifier". The IOU was set to 0.5, and the specific preprocessing steps and parameter conditions are listed in Table 3.

The test environment is set up as a stadium with multiple light sources, including natural light (accessed through windows) and artificial lighting (e.g., overhead lights, sidelights). The intensity and color of the light sources can be varied over time to simulate lighting conditions during different match periods.

The study uses manual collection of badminton dataset. Specifically, 15 video streams of badminton flights (with a resolution of 1280x720) are captured laterally in different scenes by a ZED camera and used to produce the badminton dataset. Each video stream has an effective duration of about 40 seconds and a frame rate of 30 frames/second. This means about 18,000 RGB three-channel flying badminton images of 1280x720x3 size are used for model training and testing (the badminton resolution is distributed from 20x20 to 40X40 pixels, which is a small target). The shooting scenes include the outer wall of the engineering hall with various colour backgrounds, laboratory, car park, woods, roadside, etc. They are classified into simple and complex scenes according to the backgrounds, lighting, interferences and other factors. An additional 500 static badminton images from different scenes are collected. The dataset is partitioned in a ratio of 2: 1. About 12,000 images (10 video streams) are used as the training set and the remaining 6,000 images (5 video streams of badminton flights on the orange wall, blue wall, green wall, laboratory and car park, respectively) are used as the test set, with no overlap between the training set and the test set.

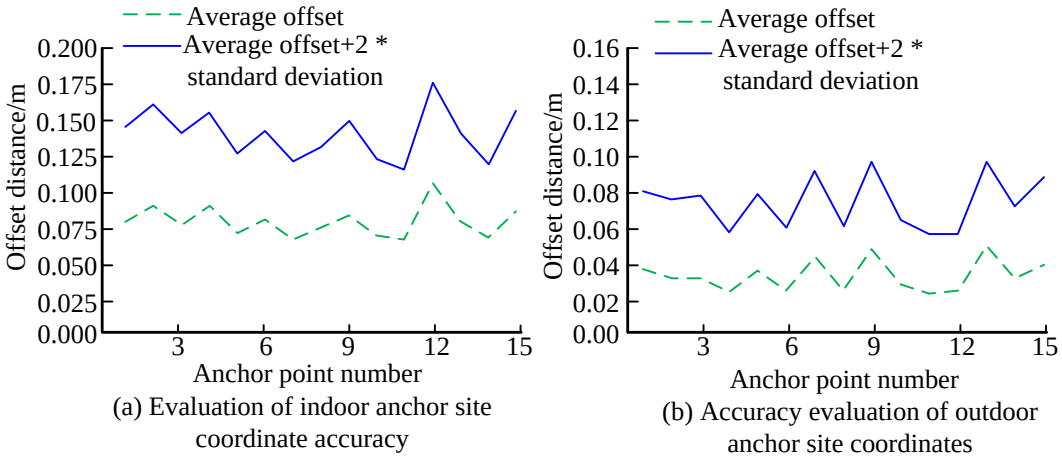
Table 3: Badminton video data preprocessing and parameter settings

Processing algorithm	Parameter settings	Other parameters	Specific settings
Image binaryzation	The grayscale threshold is 100 pixels	Badminton pixel range	20×20 pixels~40×40 pixels
Median filtering	3×3 square kernel	IOU	Zero point five
Corrosion	3×3 square kernel	Duration of each video sample	40s
Expand	12×12 square kernel	Video sample frame rate	30 frames per second
Background of the venue	Red, green, etc	Video sample RGB	1280×720×3

3.2 Performance analysis of badminton video trajectory tracking recognition and landing point prediction

Fig.7 shows the accuracy evaluation results of anchor point coordinates indoors and outdoors. The offset distance is above 0.75m for indoor badminton courts and between 0.25m-0.05m for outdoor courts. After adding twice the standard error to the offset distance, the offset distance in Fig.7 (a) is between 0.12m-0.175m, and in Fig.7 (b) it is between 0.06m-0.10m. Therefore, the accuracy of anchor coordinate conversion is better outdoors.

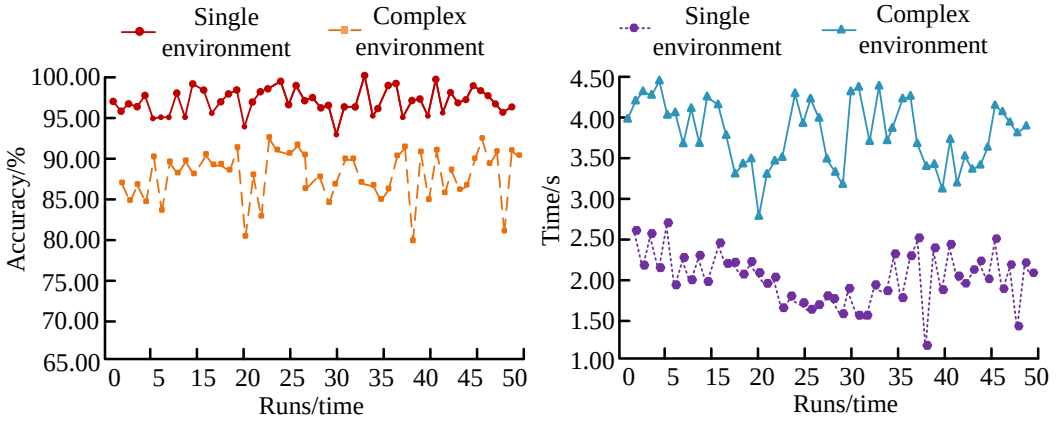
The experiment tracks the trajectory of badminton videos with different levels of complexity backgrounds. Fig.8 shows the accuracy and time results. In a single environment, the accuracy can reach over 95% with a minimum of around 87%, while in complex environments, the highest accuracy is around 92% with a minimum of around 80%. In terms of runtime, the runtime in a single environment range from 1 second to 2.8 seconds, while in complex environments it ranges from 2.5 seconds to 4.5 seconds. Therefore, the accuracy and operational efficiency of the research method are good.



(a) Evaluation of indoor anchor site coordinate accuracy

(b) Accuracy evaluation of outdoor anchor site coordinates

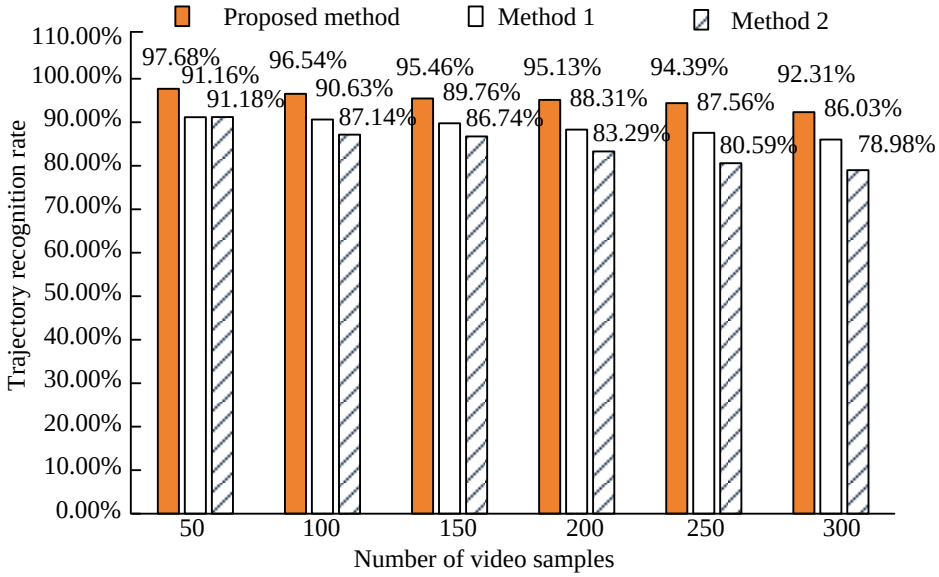
Figure 7: Evaluation results of accuracy of indoor and outdoor anchor point coordinates



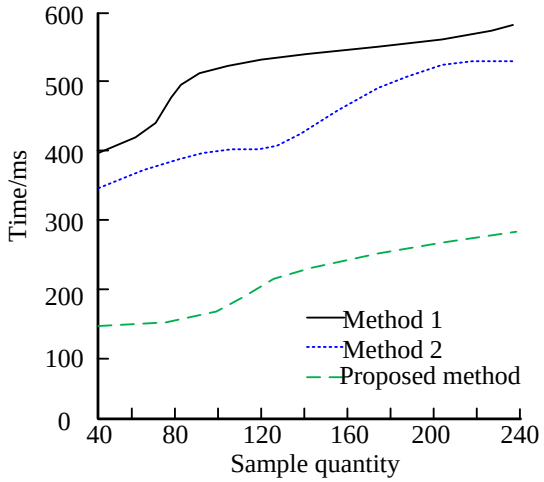
(a) Accuracy variation chart

(b) Time variation chart

Figure 8: Accuracy and algorithm running time of badminton trajectory tracking in different complexity backgrounds



(a) Different methods for trajectory recognition rate



(b) Different methods for trajectory recognition time

Figure 9: Comparison of trajectory recognition rates and recognition times using different methods

Fig.9 compares the trajectory recognition rates and recognition times of different methods. Comparison method 1 is a CV-based badminton path tracking algorithm for analyzing the TSoB proposed by Y Lyu and S Zhang. Comparison method 2 is the method of predicting the trajectory of a ball by predicting its flight path constructed by Sadegh et al. In Fig.9 (a), as the video samples increase from 50 to 300, the trajectory recognition rates of each method slightly decrease. The research method decreases from 97.68% to 92.31%, a decrease of 5.37%. The other two methods decrease by 5.13% and 12.2%, respectively, to 83.03% and 78.98%. This indicates that the research method has a better trajectory recognition rate. In Fig.9 (b), under the same sample training, the research method has a better running time, between 150-280ms, while the other two comparison methods are above 350ms and 400ms, respectively.

To analyze the prediction of badminton landing points, this study analyzes different angles of the court, fixes the camera at different points on the court, predicts the landing points, and compares them with the actual landing points in the video. After capturing the scene with three sets of cameras, the landing point obtained through the proposed Kalman prediction algorithm and the comparison algorithm (binomial prediction method) is close to the actual landing point. However, the landing point obtained by research algorithm is closer.

Table 5 shows the analysis of the low-cost running time results for badminton. During different stages of operation, the average running time of the algorithm is 621.25ms, the main coordinate calculation takes an average of 16.75ms, and the average time for trajectory prediction is 342.875ms. Their standard deviations are 12.277, 0.782, and 4.552.

Table 4: Comparison of coordinate point accuracy of different landing point prediction algorithm

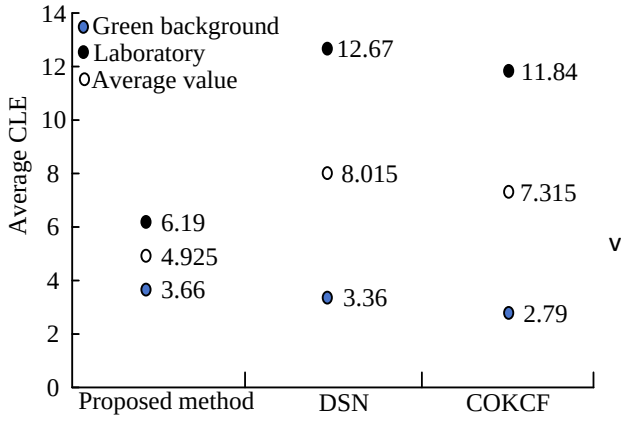
Number/Group	A		B		C	
	Pixel coordinate points	Camera coordinate points	Pixel coordinate points	Camera coordinate points	Pixel coordinate points	Camera coordinate points
Point 1	(-410,177)	-1648.623,3855, -1005.484	(-418,121)	-1604.843,3862,1166.639	(-461,159)	-1477.292,3821, -1041.031
Point 2	(-442,186)	-1378.647,3424, -871.876	(-447,120)	-1445.138,3635, -1098.457	(-491,183)	-1213.589,3332, -851.140
Point 3	(-478,184)	-1105.235,3010, -767.918	(-471,133)	-1222.846,3239,951.432	(-552,186)	-966.571,3015, -736.963
Point 4	(-523,178)	-966.642,2821, -734.415	(-527,137)	-1049.678,3086, -839.512	(-552,192)	-845.524,2753, -699.096
Point 5	(-567,184)	-829.286,2658, -671.963	(-554,198)	-905.942,2816, -711.534	(-593,214)	-731.857,2511, -557.885
Point 6	(-584,195)	-684.962,22284,566.599	(-554,181)	-790.075,2484, -651.212	(-642,281)	-553.067,2171, -398.176
Binomial	(-463.369,1995.285, -217.489)		(-386.694,1931.439, 253.585)		(-372.208,1984.073, -174.746)	

prediction point			
Kalman prediction point	(-551.604,1471.701, -376.901)	(-557.992,1667.693, -458.624)	(-493.001,1451.036, 365.496)

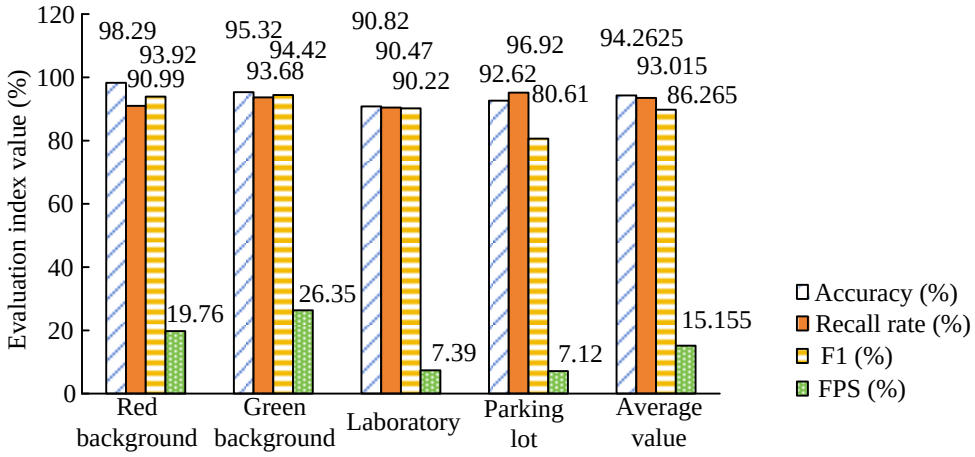
Table 5: Analysis of low-cost and time-consuming operation results of badminton

Running time	Object detection/ms	Data collection/ms	Coordinate calculation/ms	Trajectory prediction/ms	Data transmission/ms	Total testing time/ms
Run Time 1	33	201	16	235	26	611
Run time 2	33	206	19	352	22	632
Run time 3	36	205	18	345	23	627
Run time 4	34	206	18	345	23	626
Run time 5	32	207	15	340	22	616
Running time 6	34	204	17	342	23	620
Running time 7	34	207	16	338	22	617
Running time 8	33	204	15	346	23	621
Average value	33.625	205	16.75	342.875	23	621.25
Standard deviation	1.167	1.393	0.782	4.552	0.782	12.277

To ensure the reliability in badminton tracking scenarios, the experiment compares the performance of the AdaBoost-based badminton tracking algorithm with other typical tracking algorithms. These include deep learning tracking algorithm Dynamic Siamese Network (DSN) and Co-trained Kernelized Correlation Filter (COKCF). The experiment tests each video stream using three methods and calculates the average values of indicators including recall, accuracy, FPS, and F1. Four badminton flight video streams with different backgrounds are tested. The experiment compares the Central Location Error (CLE) and tracking results in different badminton scenarios. Fig.10 (a) shows a comparison of CLE. The research method has relatively small errors in different site backgrounds, at 6.19, 4.925, and 3.66, while the positioning error of DSN is relatively large (12.67, 8.015, and 3.36). Fig.10 (b) shows the tracking results of the research algorithm in multiple scenarios. In the red scene, the accuracy, recall, F1, and Frame Per Second (FPS) of the research method are 98.29%, 93.92%, 90.99%, and 19.76FPS, respectively. In relatively complex scenarios such as parking lots, the accuracy, recall, F1, and FPS of the research method are 92.62%, 96.92%, 80.61%, and 7.12FPS, respectively. The overall average accuracy, recall, F1, and FPS are 94.26%, 93.015%, 86.265%, and 15.155FPS.



(a) The proposed method and classical tracking algorithm average CLE/pixel



(b) Badminton tracking results in multiple scenarios

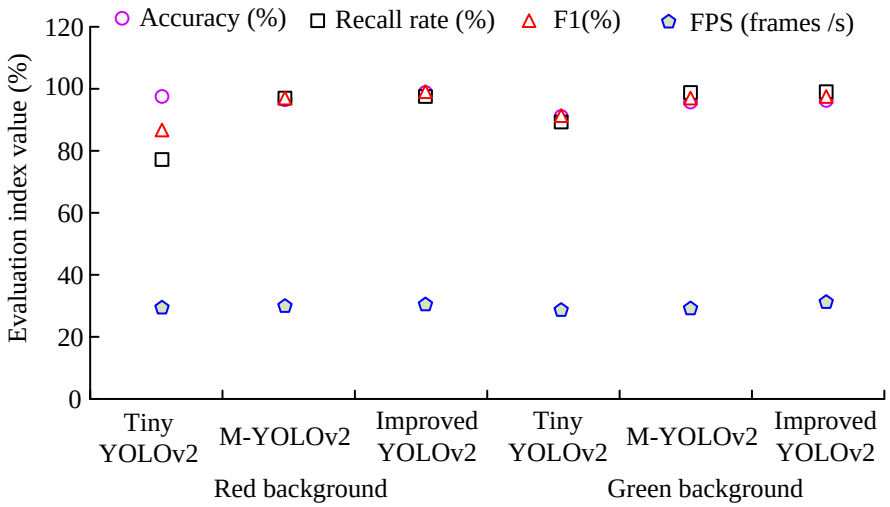
Figure 10: Performance comparison between AdaBoost-based badminton tracking algorithm and typical tracking algorithms

Table 6 presents the standard deviation of the various algorithms across 100 runs. This metric provides insight into the dispersion of the mean of a set of data, offering a measure of the stability and accuracy of the algorithms in this optimization process. The data in the table indicates that GA exhibits the poorest performance in terms of stability and accuracy in this optimization. In comparison, the PSO algorithm and CS demonstrate superior outcomes. Traditional CSA and research methods are also observed to perform better. In addition, the complexity of various computational intelligence algorithms can be reflected by the average running time and space complexity of various algorithms to complete one optimization process. In Table 6, CS is significantly larger than the other four algorithms in terms of average running time and high algorithm complexity, while the other four algorithms are closer. To sum up, the research method can be found to be superior in this optimization work after comprehensive evaluation.

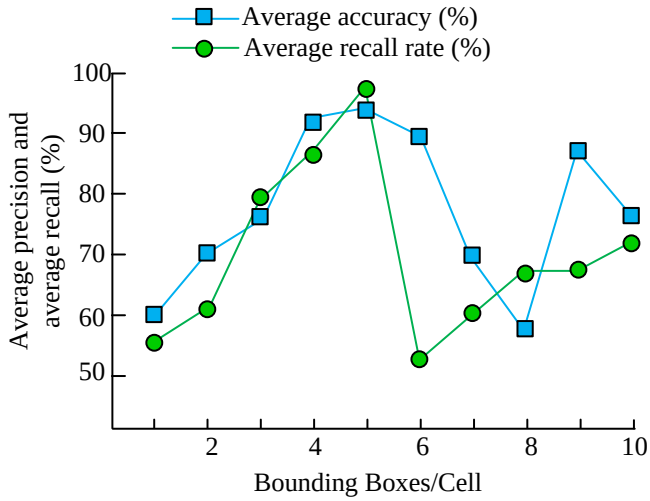
The experiment compares the performance of different YOLOv2 and verifies the advantages of the improved YOLOv2. In the red background of Fig.11 (a), the improved YOLOv2 has good performance, with accuracy, recall, F1, and FPS of 98.6%, 97.6%, 98.7%, and 30.8 FPS, among which accuracy, F1, and FPS have advantages. In green scenes, the accuracy, recall, F1, and FPS are 95.7%, 98.7%, 96.8%, and 31.4 FPS. Tiny YOLOv2 has the lowest performance, while M-YOLOv2 only has an advantage in recall rate and F1 in green background. Fig.11 (b) shows the average accuracy and recall of the improved YOLOv2 bounding box, where the average accuracy and recall in cell 5 are as high as 98.56% and 95.86%.

Table 6: Performance comparison of 5 algorithms

Algorithm type	Optimal	Worst	Mean value	Standard deviation	Mean motion time
Research Method	0.02	0.1	0.05	0.03	1360.12
CSA	0.1	0.18	0.15	0.03	1327.23
CS	0.12	0.17	0.14	0.04	5458.25
GA	0.2	0.51	0.32	0.09	1300.95
PSO	0.28	0.41	0.34	0.05	1436.61



(a) Experimental results of YOLO series network badminton detection in different scenarios



(b) Research on the average precision and recall of improved YOLOv2 bounding boxes

Figure 11: Performance comparison of different YOLOv2

3.3 Discussion

The results of the comparison with state-of-the-art (SOTA) indicate that the trajectory recognition rate of each method exhibits a slight decline as the number of video samples increases from 50 to 300. The research method demonstrates a reduction from 97.68% to 92.31%, representing a 5.37% decrease, while the other two methods exhibit a decline of 5.13% and 12.2%, respectively, to 83.03% and 78. The research method achieves a trajectory recognition rate of 98%, demonstrating superior performance compared to the other methods. The introduction of a noise suppression mechanism and the use of KF to smooth the data enables the research method to effectively reduce the impact of noise on the prediction results. The detection of badminton employs an enhanced YOLO algorithm to identify the speed and accuracy of badminton, thereby obtaining precise coordinates. In the same sample training, the proposed method has a better running time between 150-280ms, while the other two comparison methods are above 350ms and 400ms, respectively. This is due to the fact that the research method employs a dynamic adaptation or online learning mechanism, which allows the model to continuously learn and update its own parameters in real applications to adapt to environmental changes and emerging challenges. Moreover, the study employs a more advanced CNN in deep learning, which can automatically learn and extract features from the raw data that are more critical for trajectory prediction. In contrast, the parameters of the existing comparison methods are hand-designed and these features are not robust enough in complex environments. The research method has been deeply optimized in many aspects such as algorithm design,

experimental conditions, and environmental adaptability. Together, these advantages enable the research method to improve the prediction accuracy in different environments and show stronger robustness and generalization ability.

4 Conclusion

Badminton is a popular sport, and its flight trajectory is an important factor in badminton competitions. To effectively identify badminton trajectories and predict landing points, this study proposed the FCST method, which combines MRES. Then, based on the video stream, badminton trajectory recognition and landing point prediction were carried out, and an improved YOLO algorithm was designed to improve recognition accuracy. LSM was used for trajectory fitting to solve the trajectory curve and landing point prediction. Research has shown that as the number of video samples increased from 50 to 300, the trajectory recognition rates of various methods slightly decreased. The research method decreased from 97.68% to 92.31%, a decrease of 5.37%. The other two methods decreased by 5.13% and 12.2%, respectively, and decreased to 83.03% and 78.98%. This indicated that the research method had a better trajectory recognition rate. Under the same number of training samples, the research method had a better running time, ranging from 150-280ms, while the comparison method had a running time of 350ms and above 400ms. The research method had relatively small errors in different site backgrounds (6.19, 4.925, 3.66), while the positioning error of the DSN was relatively large (12.67, 8.015, 3.36). The overall average accuracy, recall, F1, and the FPS of the research method were 94.26%, 93.015%, 86.265%, and 15.155FPS. The above data indicate that the research method can identify the trajectory curve of badminton and predict the landing point coordinates. Nevertheless, the research on the badminton in flight, including the resistance and other factors affecting the consideration, is insufficient. Consequently, subsequent research should be optimized based on three-dimensional reconstruction algorithms. This can be achieved by combining badminton resistance and other factors with depth of information, thereby obtaining the badminton three-dimensional coordinates of the point of prejudgement. Concurrently, the algorithmic approach enables the comprehensive and detailed statistical analysis, thereby facilitating the advancement of a streamlined and effective badminton robotic vision system. The system is designed to be a straightforward and efficient vision system for badminton robots.

References

[1] R. Prakash, L. Behera, S. Mohan, and S. Jagannathan, "Dynamic trajectory generation and a robust controller to intercept a moving ball in a game setting," IEEE Transactions on Control Systems Technology, vol. 28, no. 4, pp. 1418-1432, 2020. https://doi.org/10.1109/TCST.2019.2913129

[2] S. A. Maged, H. E. A. E. Munim, Y. M. Abdelkhalek, and M. I. Awad, "Trajectory based fast ball detection and tracking for an autonmous industrial robot system," International Journal of Intelligent Systems Technologies and Applications, vol. 20, no. 3, pp. 126-145, 2021. https://doi.org/10.1504/IJISTA.2021.10038044

[3] X. Wang, "Badminton trajectory tracking based on D-H Kinematics Model," 2020 International Conference on Data Processing Techniques and Applications for Cyber-Physical Systems, vol. 2021, no. 2021, pp. 465-474, 2021. https://doi.org/10.1007/978-981-16-1726-3_57

[4] B. A. Costa, F. L. Parente, J. S. Belfo, P. I. NicolaRosa, J. L. Jose, M. Belhadj, and M. Joao, "A reinforcement learning approach for adaptive tracking control of a reusable rocket model in a landing scenario," Neurocomputing, vol. 577, no. 7, pp. 1-15, 2024 https://doi.org/10.1016/J.NEUCOM.2024.127377

[5] L. Zhu, "A prediction method for the service trajectory of badminton moving video based on fuzzy clustering algorithm," International Journal of Innovative Computing and Applications, vol. 12, no. 4, pp. 216-223, 2021. https://doi.org/10.1504/IJICA.2021.10038620

[6] G. Cui, B. Zhang, and R. Marlene, "Trajectory simulation of badminton robot based on fractal brown motion," Fractals, vol. 28, no. 8, pp. 1-11, 2020. https://doi.org/10.1142/S0218348X20400216

[7] Y. Lyu and S. Zhang, "Badminton path tracking algorithm based on computer vision and ball speed analysis," Hindawi, vol. 2021, no. 10, pp. 1-13, 2021. https://doi.org/10.1155/2021/3803387

[8] T. Lin, A. Aouididi, Z. Chen, J. Beyer, H. Pfister, and J. H. Wang, "VIRD: Immersive match video analysis for high-performance badminton coaching," IEEE Transactions on Visualization and Computer Graphics, vol. 30, no. 1, pp. 458-468, 2024. https://doi.org/10.1109/TVCG.2023.3327161

[9] S. Vial, J. L. Croft, R. T., Schroeder, A. J. Blazevich, and J. C. Wilkie, "Does the presence of an opponent affect object projection accuracy in elite athletes? A study of the landing location of the short serve in elite badminton players," International Journal of Sports Science & Coaching, vol. 15, no. 3, pp. 412-417, 2020. https://doi.org/10.1177/1747954120915670

[10] A. Sadegh, D. Vali, and J. Fatemeh, "Tennis ball trajectory estimation using GA-based fuzzy adaptive nonlinear observer," International Journal of Dynamics and Control, vol. 10, no. 5, pp. 1685-1696, 2022. https://doi.org/10.1007/s40435-022-00921-9

[11] Y. Gu, D. Gao, and J. Yang, "A design model of target-hitting trajectory drilled into the reservoir with elliptical truncated cone surface," Chemistry

and Technology of Fuels and Oils, vol. 57, no, 1, pp. 107-119, 2021. https://doi.org/10.1007/s10553-021-01231-0

[12] C. Cuiping, "Badminton video analysis based on player tracking and pose trajectory estimation," 2021 13th International Conference on Measuring Technology and Mechatronics Automation (ICMTMA), vol. 2021, no. 13, pp. 471-474, 2021. https://doi.org/10.1109/ICMTMA52658.2021.00108

[13] A. Umek, and A. Kos, "Validation of UWB positioning systems for player tracking in tennis," Personal and Ubiquitous Computing, vol. 26, no. 6, pp. 1023-1033, 2020. https://doi.org/10.1007/s00779-020-01486-0

[14] A. G. Melo, M. F. Pinto, A. L. M. Marcato, I. Z. Biundini, and N. M. S. Rocha, "Low-cost trajectory-based ball detection for impact indication and recording," Journal of Control, Automation and Electrical Systems, vol. 32, no. 2, pp. 367-377, 2021. https://doi.org/10.1007/s40313-020-00677-7

[15] H. Zhao, and F. Hao, "Target tracking algorithm for table tennis using machine vision," Journal of Healthcare Engineering, vol. 2021, no. 4, pp. 1-7, 2021. https://doi.org/10.1155/2021/9961978

[16] X. Yu, G. Xiang, H. Collopy, and X. Kong, "Trajectory prediction of a model rocket falling into the towing tank: Experimental tests versus numerical simulations," Journal of Aerospace Engineering, vol. 33, no. 5, pp. 1-14, 2020. https://doi.org/10.1061/(ASCE)AS.1943-5525.0001172

[17] H. Yu, X. Liang, J. Han, and Y. Fang, "Adaptive trajectory tracking control for the quadrotor aerial transportation system landing a payload onto the mobile platform," IEEE Transactions on Industrial Informatics, vol. 20, no, 1, pp. 23-37, 2024. https://doi.org/10.1109/TII.2023.3256374

[18] R. Devi, "Flight trajectory precision in rocketry: A case study on vertical landing," IOSR Journal of Electron Communications Engineering, vol. 9, no. 14, pp. 9-14, 2023. https://doi.org/10.9790/2834-1806010914

[19] T. Pan, "Tracking and extracting action trajectory of athlete based on hierarchical features," Ingénierie des Systèmes d'Information., vol. 25, no. 5, pp. 677-682, 2020, https://doi.org/10.18280/isi.250515

[20] J. Hyo-Geon, C. H. Hyun, and B. Park, "Virtual angle-based adaptive control for trajectory tracking and balancing of ball-balancing robots without velocity measurements," International Journal of Adaptive Control and Signal Processing, vol. 37, no. 6, pp. 2204-2215, 2023. https://doi.org/10.1002/acs.3634

[21] H. X. Li, F. Y. Gao, C. J. Hu, Q. L. An, X. Q. Peng, and Y. M. Gong, "Trajectory track for the landing of carrier aircraft with the forecast on the aircraft carrier deck motion," Mathematical Problems in Engineering, vol. 2021, no. 52, pp. 1-11, 2021. https://doi.org/10.1155/2021/5597878

[22] T. Rogalski, P. Rzucidlo, S. Noga, and D. Nowak, "The use of vision system to determine lateral deviation from landing trajectory," Aircraft Engineering and Aerospace Technology, vol. 95, no. 9, pp. 1363-1377, 2023. https://doi.org/10.1108/AEAT-10-2022-0265

[23] C. Lai, X. Lin, Z. Su. and D. W. Zhao. "Event-triggered coordinated trajectory tracking control for multiple fully-actuated MSVs under prescribed performance constraints and limited transmission resources," Ocean Engineering, vol. 281, no. 8, pp. 1-13, 2023. https://doi.org/10.1016/j.oceaneng.2023.114529

[24] V. A. Zubko, N. A. Eismont, K. S. Fedyaev, and A. A. Belyaev, "A method for constructing an interplanetary trajectory of a spacecraft to Venus using resonant orbits to ensure landing in the desired region," Advances in Space Research, vol. 72, no. 2, pp. 161-179, 2023. https://doi.org/10.1016/j.asr.2023.02.045

[25] B. Wang, H. Lin, C. Tang, and G. Xu, "Autonomous deck landing of a vertical take-off and landing unmanned aerial vehicle based on the tau theory," Transactions of the Institute of Measurement and Control, vol. 45, no. 2, pp. 233-248, 2022. https://doi.org/10.1177/01423312221104424

[26] H. Wang, J. Yang, M. Hu, J. Tang, and W. Yu, "A comparative analysis for eye movement characteristics between professional and non-professional players in FIFA eSports game," Displays, vol. 81, no. 1, pp. 1-11, 2024. https://doi.org/10.1016/j.displa.2023.102599

[27] R. I. Popescu, M. Raison, G. M. Popescu, S. David, and S. Achiche, "Design and development of a novel type of table tennis aerial robot player with tilting propellers," Mechatronics, vol. 74, no. 4, pp. 1-3, 2021. https://doi.org/10.1016/j.mechatronics.2021.102483

[28] A. J. Radcliffe, and G. V. Reklaitis, "Automated object tracking, event detection and recognition for high log peed video of drop formation phenomena," AIChE Journal, vol. 67, no. 8, pp. 1-20, 2021. https://doi.org/10.1002/aic.17245

[29] K. S. Yadav, K. Anish Monsley, R. H. Laskar, S. Misra, M. K. Bhuyan, and T. Khan, "A selective region-based detection and tracking approach towards the recognition of dynamic bare hand gesture using deep neural network," Multimedia Systems, vol. 28, no. 3, pp. 861-879, 2022. https://doi.org/10.1007/s00530-022-00890-1

[30] F. Shang, P. Yang, J. Xiong, Y. Feng, and X. Li, "Tamera: Contactless commodity tracking, material and shopping behavior recognition using COTS RFIDs," ACM Transactions on Sensor Networks, vol. 19, no. 2, pp. 1-24, 2023. https://doi.org/10.1145/3563777

[31] K. Bhosle, and V. Musande, "Evaluation of deep

learning CNN Model for recognition of devanagari digit," Artificial Intelligence and Applications, vol. 1, no. 2, pp. 114-118, 2023. https://doi.org/10.47852/bonviewAIA3202441

[32] J. Zhang, S. Yu, Y. Yan, and Y. Zhao, "Fixed-time sliding mode trajectory tracking control for marine surface vessels with input saturation and prescribed performance constraints," Nonlinear Dynamics, vol. 112, no. 19, pp. 17169-17181, 2024 https://doi.org/10.1007/s11071-024-09918-9

[33] F. Koepf, S. Kille, J. Inga, and S. Hohmann, "Adaptive optimal trajectory tracking control applied to a large-scale ball-on-plate system," American Control Conference, vol. 2021, no. 5, pp. 1352-1357, 2021. https://doi.org/10.23919/ACC50511.2021.9482629

[34] H. Ni, "Face recognition based on deep learning under the background of big data," Informatica, vol. 44, no. 4, pp. 491-495, 2020. https://doi.org/10.31449/INF.V44I4.3390.