



# BASKETBALL PLAYER TARGET TRACKING BASED ON IMPROVED YOLOV5 AND MULTI FEATURE FUSION

Jinjun Sun<sup>1</sup>  and Ronghua Liu<sup>2,\*</sup> 

<sup>1</sup>*Department of Safety and Security, Zhejiang Posts and Telecom College, Shaoxing, China*

<sup>2</sup>*Department of Fundamental Discipline, Department of Physical Education,  
Shanghai University of Finance and Economics, Zhejiang College, Jinhua, China*

*\*Corresponding author: Ronghua Liu (liu2135173@163.com)*

Submitted: 18 Nov 2024 Accepted: 8 Jan 2025 Published: Mar 20, 2025

Licence: CC BY-NC 4.0 

**Abstract** Multi-target tracking has important applications in many fields including logistics and transportation, security systems and assisted driving. With the development of science and technology, multi-target tracking has also become a research hotspot in the field of sports. In this study, a multi-attention module is added to compute the target feature information of different dimensions for the leakage problem of the traditional fifth-generation single-view detection algorithm. The study adopts two-stage target detection method to speed up the detection rate, and at the same time, recursive filtering is utilized to predict the position of the athlete in the next frame of the video. The results indicated that the improved fifth generation monovision detection algorithm possessed better results for target tracking of basketball players. The running time was reduced by 21.26% compared with the traditional fifth-generation monovision detection algorithm, and the average number of images that could be processed per second was 49. The accuracy rate was as high as 98.65%, and the average homing rate was 97.21%. During the tracking process of 60 frames of basketball sports video, the computational delay was always maintained within 40 ms. It can be demonstrated that by deeply optimizing the detection algorithm, the ability to identify and locate basketball players can be significantly improved, which provides a solid data support for the analysis of players' behaviors and tactical layout in basketball games.

**Keywords:** YOLOv5, object detection, action characteristics, recursive filtering, Mahalanobis distance, Hungarian algorithm.

## 1. Introduction

In the field of sports competition, basketball possesses the characteristics of high-speed confrontation and precise cooperation. In-depth analysis of athletes' performance is the key to improve team tactics and individual skills [24]. With the rapid development of computer vision and artificial intelligence technology, algorithms are gradually applied to target tracking (TT) of basketball players, showing great potential [4]. Through high-precision image processing and intelligent recognition technology, the algorithms are able to track key information such as the position, speed and movement trajectory of each player on the court in real time. This provides coaching teams with unprecedented game insight data [27]. This not only changes the way of training, but also promotes the scientific development of game strategies, enabling basketball to move towards a smarter and more efficient future in the wave of digitization.

In terms of target detection, Song et al. designed an intelligent recognition system combining multi-TT algorithm and YOLOv5 ware in order to solve the problem of

fine target occlusion affecting helmet detection. The actual test results at a complex construction site indicated that the average accuracy of the intelligent recognition system was 94.5%, and the detection speed was up to 40 fps, which basically realized the real-time detection [21]. Zhan et al. improved the algorithm's target detection performance in UAV scenarios and chose to incorporate four methods to improve small target detection accuracy based on YOLOv5. The findings demonstrated that the model that combined the several improvement techniques not only significantly increased detection accuracy but also successfully decreased detection speed loss up to 55 fps [32]. Bharathi and Anandharaj developed a YOLOv5 multi-TT model that could detect, track and recognize individuals in order to help surveillance cameras measure social distances in road traffic videos. The results of the study found that the model achieved good results with 93% precision, 94% recall and 95% all class average precision for measuring social distance by object classification and localization in real time traffic surveillance video [1].

In terms of real-time tracking of motion trajectory, Hao et al. used the maximum interclass variance method for grayscale feature processing in an attempt to solve the efficiency problems of the current algorithms related to athlete detection and recognition. The study was based on Harris corner extraction algorithm and proposed multi-TT combining target corner features. The study showed that the algorithm performed well and had some practical effects [9]. Facchinetti et al. proposed an algorithm to automatically identify the active period of the sport using the tracking data of the athletes in basketball in order to obtain the accurate data of basketball between the course of the game and the intermission [5]. A basketball, a basket, and athletes were the feature extraction (FE) objects in Wang and basketball sports video TT method, which they combined with an upgraded gray neural network technique to better assess the condition of athletes in the video. The approach could successfully and accurately identify basketball movements, according to the findings of experimental testing, offering a new technique for basketball movement detection [25].

In summary, in the field of target detection and motion trajectory tracking, although some progress has been made in existing researches, such as improving detection accuracy and real-time performance, the detection efficiency is not high, and it is easy to miss detection in the case of target overlap and occlusion. Based on this, the research creates a new enhanced visual inspection algorithm (improved you only look once version 5, I-YOLOv5). To solve the missed detection problem caused by overlapping targets, a multi-attention module (AM) is innovatively added, and recursive filtering is used to predict the next position of the player, and then the prediction results are corrected by the actual situation.

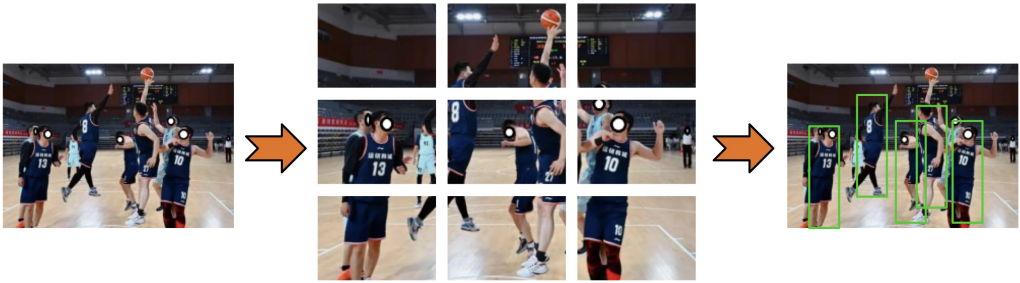


Fig. 1. Detection of basketball players by YOLOv5.

## 2. Methods

### 2.1. I-YOLOv5 algorithm for adding an attention module

With the increasing demand for sports event analysis and automated referee systems, computer detection plays an important role in real-time monitoring, action recognition, event detection, and content understanding. The traditional YOLOv5 algorithm is widely used in object detection due to its excellent real-time performance and high detection accuracy. However, the YOLOv5 algorithm is prone to inaccurate and undetected targets with small volume and high density, especially when dealing with complex backgrounds, athlete occlusions, and rapid movements, which may encounter problems such as missed detections, false detections, or unstable tracking [33]. The detection process of YOLOv5 algorithm for basketball players is shown in Figure 1. In this Figure, the YOLOv5 algorithm for the detection of the target, is required to cut the sports scene into multiple parts before detection, two for the absence of human body features of the block does not do the detection, which can reduce the algorithm's resource usage. However, for the basketball sports scene with more personnel, some basketball players only show part of their bodies due to the overlapping occlusion of personnel. The feature details are easily erased after the cutting process, resulting in a missed detection during the tracking detection process. For example. There should have been 7 people in the scene, but it is omitted to be detected as 5 people. For this reason, I-YOLOv5 is created by improving the YOLOv5 algorithm. In the I-YOLOv5 algorithm, a multi-AM is added to model the multi-dimensional situation simultaneously, and the information of different dimensions is fully displayed. The necessity of Global Average Pooling (GAP) lies in its ability to effectively aggregate feature map information into a global information representation, thereby reducing the number of model parameters, lowering computational complexity, and improving the model's generalization ability. The specific calculation method of GAP is to sum up all pixel values of the feature map and then divide by the total number of pixels. In the multi-AM, the introduction of GAP helps to enhance

the model's attention to important features, improve the quality and diversity of feature representation. The formula for GAP in multi-AM is shown in Equation (1) [26].

$$\text{CA}(T_x) = \frac{\sum_{a=0}^{K-1} \sum_{b=0}^{G-1} T_x(a, b)}{K \times G}, \quad (1)$$

where  $\text{CA}(T_x)$  represents the global information under the  $x$ th channel of the image,  $T$  represents the image,  $(a, b)$  represents the coordinate point position,  $a$  represents the width coordinate,  $b$  represents the height coordinate,  $K$  represents the image width, and  $G$  represents image height. The global information is obtained using only the basic features of the image without adding other information. To make the channel information richer and to obtain more representative global information, the 2D discrete cosine transform is combined to obtain channel information of more frequency bands, which is used to enrich the global information. The computation of the 2D discrete cosine transform spectrum is demonstrated by Equation (2) [31].

$$P_{k,g}^{2D} = \sum_{k=0}^{K-1} \sum_{g=0}^{G-1} \text{In}_{a,b}^{2D} Q_{k,g}^{a,b}, \quad (2)$$

where  $P_{k,g}^{2D}$  represents the two-dimensional discrete cosine (2D-DC) transform spectrum under the height dimension frequency score  $g$  and width dimension frequency score  $k$ ,  $\text{In}_{a,b}^{2D}$  represents the two-dimensional input parameters,  $Q_{k,g}^{a,b}$  represents the weight score, and  $D$  represents dimension. The weighting score is calculated as shown in Equation (3) [16].

$$Q_{k,g}^{a,b} = \cos\left(\frac{\pi g(a+0.5)}{G}\right) \times \cos\left(\frac{\pi k(b+0.5)}{K}\right), \quad (3)$$

where both the height-dimensional frequency score  $g$  and the width-dimensional frequency score  $k$  are 0, and  $Q_{k,g}^{a,b} = 1$ . The relation between the 2D-DC transform spectrum and the GAP is shown in Equation (4).

$$P_{K_w}^{2D} = \sum_{k=0}^{K-1} \sum_{g=0}^{G-1} \text{In}_{a,b}^{2D} = K \times G \times \text{CA}(T_x), \quad (4)$$

where  $T_x$  represents the  $x$  channel of image  $T$ . At this point the 2D discrete cosine transform spectrum and the global mean pool are in a positive correlation. The same applies for frequency scores of different dimensions, and feature information of many different dimensions can be calculated [22]. Figure 2 depicts the structure of the channel AM.

In Figure 2, the channel AM splits the image into slices of different parts. Equation (5) specifies how each slice's channel score is determined [20].

$$P^e = \sum_{k=0}^{K-1} \sum_{g=0}^{G-1} \text{In}_{a,b}^{2D} Q_{k,g}^{a,b} = 2\text{DT}, \quad (5)$$

where  $P^e$  represents the calculated channel score, 2DT represents the 2D-DC transform. The new merged channel is formed after integrating all the sliced processed channel scores. The merging process is shown in Equation (6) [8].

$$P_Z = \text{cat}([P_1, P_2, \dots, P_n]), \tag{6}$$

where  $P_n$  represents the sliced channel parameters of different layers,  $P_Z$  represents the merged channels, and  $\text{cat}(\cdot)$  represents the merge operation. The new channels are subsequently integrated with the slices to form a new channel AM [30]. The structure of another coordinate AM in the I-YOLOv5 algorithm is shown in Figure 3. In this Figure, in order to accurately capture the key information of the image in the width and height dimensions and encode the positions, it is necessary to apply special pooling operations to the input feature map (FM) along the horizontal and vertical directions, respectively. After determining the input parameter features, the special positions of all channels in the width direction are numbered. The vertical data of the channels are calculated as

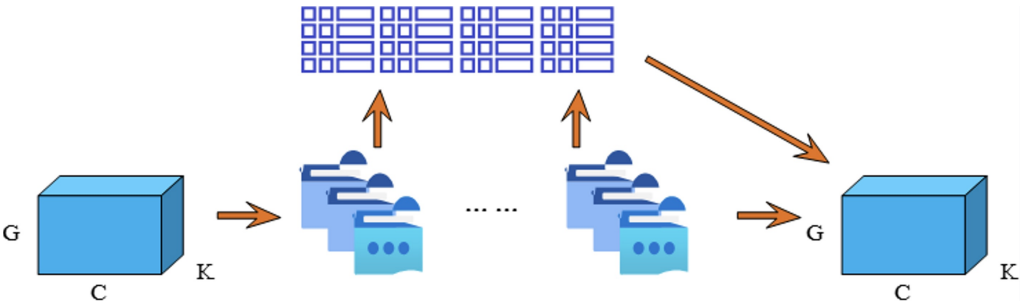


Fig. 2. Structure diagram of the channel attention module.

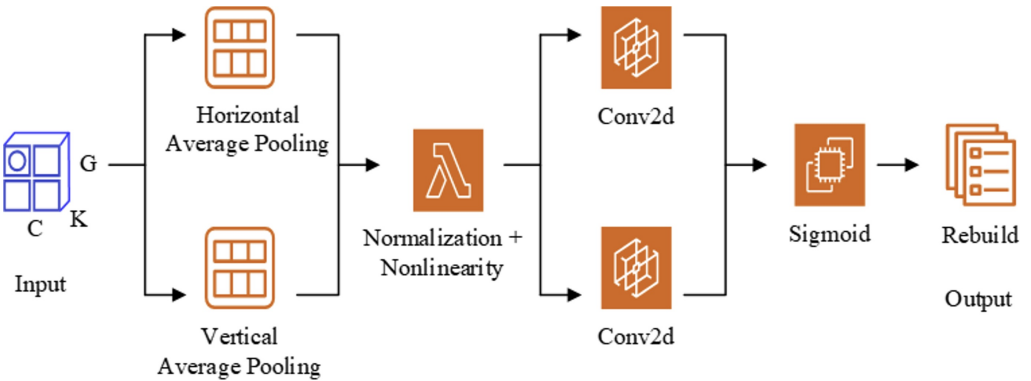


Fig. 3. Coordinate the structure of the attention module.

shown in Equation (7) [34].

$$O_x^{g_h} = \frac{\sum_{0 \leq b \leq K_w} n_x(g_h, a)}{K_w}, \quad (7)$$

where  $O_x^{g_h}$  represents the calculated value of vertical data with height  $g_h$  under the  $x$ th channel,  $(a, b)$  represents the point coordinates,  $K_w$  represents the width, and  $n_c(g_h, a)$  represents the value of the image slice with height  $g_h$  and width coordinate  $a$  under the  $x$ th channel. Similarly, the horizontal data of the channel is calculated as shown in Equation (8) [18].

$$O_x^{k_o} = \frac{\sum_{0 \leq b \leq G} n_x(b, k_o)}{G}, \quad (8)$$

where  $O_x^{k_o}$  represents the calculated value of the horizontal data representing the width of  $k_o$  under the  $x$ th channel,  $G$  represents the height, and  $n_x(b, k_o)$  represents the value of the image slice with width  $k_o$  and height coordinate  $b$  under the  $x$ th channel. These two specific operations are the core steps of feature processing. Integrating information along two different spatial dimensions, respectively, generates a pair of directionally sensitive FMs [29]. This process not only enhances the model's ability to capture long-range dependencies in one spatial dimension, but also subtly maintains precise spatial location details in the other dimension, thus optimizing the model's recognition and localization performance of the target object. With these two transformations, the model is able to analyze the spatial structure of the image or data more effectively and achieve more accurate target localization [11]. In order for the algorithm to obtain a faster running speed, the multi-AM and the coordinate AM are added to the I-YOLOv5 algorithm using a tandem approach. A brief description of the structure of I-YOLOv5 is shown in Figure 4. In this Figure the multi-AM and the coordinate AM also contain different component modules that implement the processing of the input parameters. The uniqueness of multi AM in I-YOLOv5 lies in its combination of Cross Stage Partial Network (CSP) module and Spatial Pyramid Pooling (SPP) module. The CSP module segments the input feature map, with one part passed directly and the other part merged after residual network processing to improve efficiency and feature learning. The SPP module captures multi-scale features and enhances the detection capability of multi-scale targets through parallel operations of multi-scale pooling kernels. While generating a coordinate description parameter through average pooling, the maximum pooling operation is used to obtain the maximum value of the coordinate parameter. Based on the different feature descriptions, the parameters are transferred to the data processing center module. By integrating the various parameters, the eligible feature values are produced, which enhances the I-YOLOv5 algorithm's detection performance. For the detection of basketball players, it is also necessary to input the specified features and the corresponding feature recognition structure. The current motion image recognition is only good at

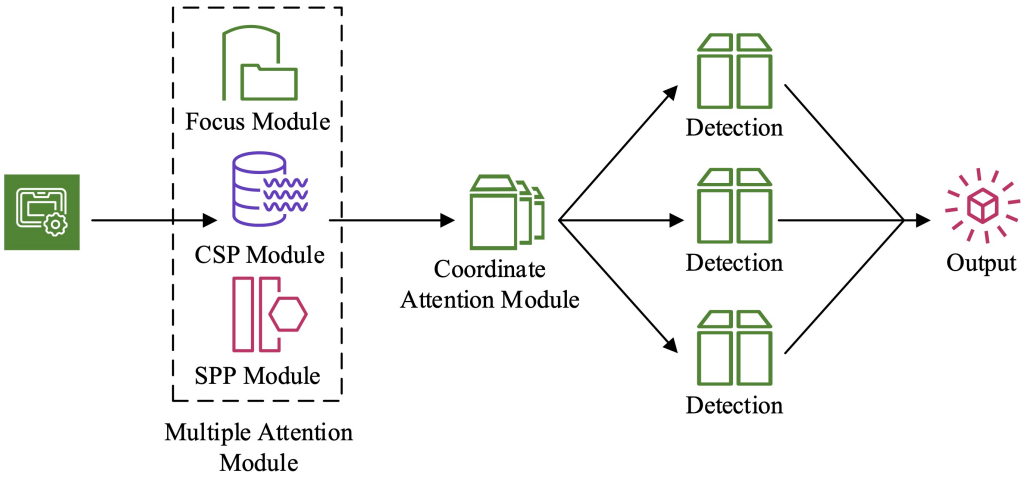


Fig. 4. I-YOLOv5 structure scheme.

tracking simple movements of a single target, while the movements of basketball players are often complex and variable. For this reason, a module for real-time discrimination of multiple features is also needed to improve the accuracy of detection.

## 2.2. Basketball player feature detection module

The key to more effectively deal with the complex and variable action recognition problem of basketball players is to construct an advanced detection module with adaptive ability and accurate target placement labeling in the image. Whether it is based on conventional algorithms or self-learning feature detection modules, the core of the effectiveness lies in whether or not the target is preset and accurately labeled in the recognition image. In the field of basketball player tracking, target detection, as a key technique, is directly related to the accuracy ratio (AR) of the tracking results [15]. Traditional target detection algorithms suffer from candidate region redundancy, high computation, low FE dependency, lack of robustness and fragmented detection process. These problems limit the detection efficiency and accuracy. In order to solve these problems, reduce computational burden and improve feature expression, fusion of detection links is needed to achieve global optimization. The two-stage target detection (TSTD) algorithm is an algorithm that provides high accuracy and is divided into two main processes. Firstly, it generates pending regions that may contain targets, and subsequently categorizes and edge-boxes recede from these pending regions. The whole process is shown in Figure 5 [10].

In Figure 5, the first stage of TSTD algorithm is the Regional suggestion network.

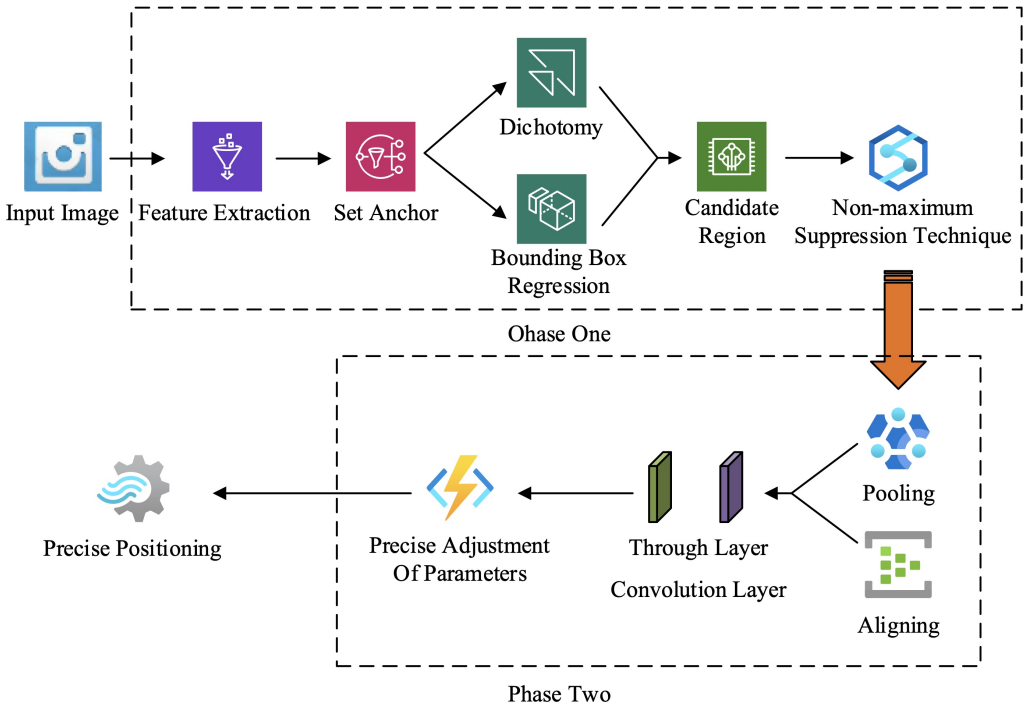


Fig. 5. Non-maximum suppression technique.

It is responsible for generating high-quality candidate regions from images [7]. The region suggestion network utilizes pre-trained convolutional neural networks to extract feature maps and places anchor points of different sizes and proportions on them. By binary classification and bounding boxes (BOBs) regression, the region suggestion network identifies anchor points that may contain targets and uses non-maximum suppression techniques to remove overlapping and low-confidence candidate regions.

The second stage of TSTD algorithm is the Classification and regression networks. The task of this stage is to refine the candidate regions generated by the region suggestion network [13]. In this stage, the candidate regions are transformed into fixed-size feature maps through region of interest pooling techniques, and then further feature extraction is performed using fully connected layers or convolutional layers. Finally, the network outputs the category probability and precise bounding box position for each candidate region. The architecture of the masked region-based CNN as a commonly used detection model in region suggestion networks is shown in Figure 6. The input image is first processed by the masked region-based CNN in Figure 6 before entering the



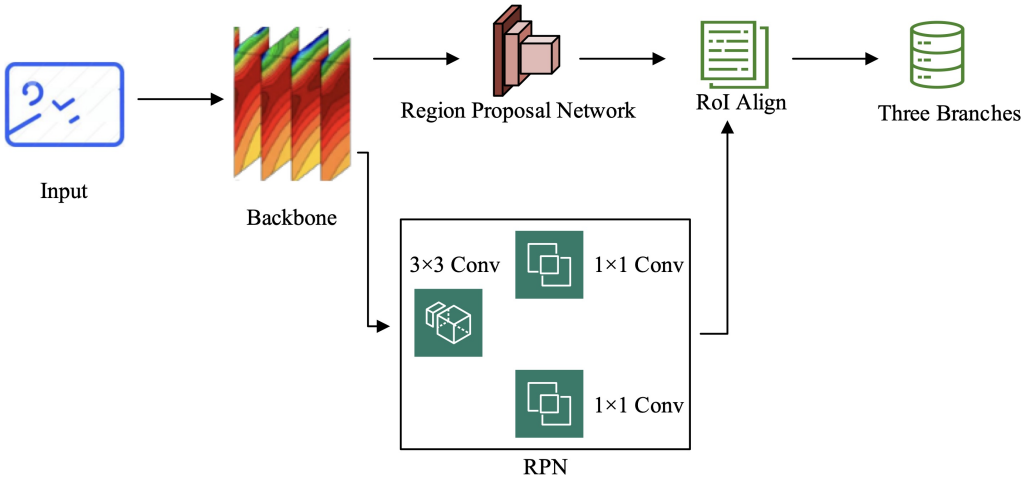


Fig. 6. Mask R-CNN structure.

classification model’s backbone network. The backbone network is used to extract FMs with high semantic content by removing fully linked layers. The FM has a certain multiplicative reduction relation with the original image, and subsequently, the FM enters the core of the mask region-based CNN (MRCNN), i.e., the region suggestion network layer. The MRCNN generates candidate regions on the original map by means of small neural networks. Each FM pixel point corresponds to multiple candidate regions of the original map. The MRCNN then predicts the coordinate offsets of these candidate regions and the probability of whether they are foreground or not by convolution. The candidate regions are adjusted and filtered to select regions that are likely to contain objects. Finally, these candidate regions are accurately mapped and adjusted through the region of interest alignment layer. Candidate regions are mapped onto the FM by interpolation and uniformly resized. It is ensured that the candidate regions contain rich information of the original map to prepare for the subsequent fine recognition [19]. The loss function (LF) of the region of interest alignment layer is calculated as shown in Equation (9).

$$L_{ROI} = L_c + L_b + L_m, \quad (9)$$

where  $L_{ROI}$  represents the LF of the region of interest alignment layer,  $L_c$  represents the classification LF,  $L_b$  represents the candidate region LF, and  $L_m$  represents the mask LF. The categorization LF mostly shows the discrepancy between the realistic categories and the predicted categories of the algorithm. The candidate region LF mainly represents the balance of the samples. The mask LF mainly indicates the loss value of the output value of each dimension. The control LF can effectively avoid the confusion of recognition of

approximate features. The second categorization process of TSTD, for each category, the candidate box with the highest score is selected first [14]. Subsequently, the intersection ratio between the remaining candidate boxes and the highest scoring candidate box is calculated. If the intersection ratio exceeds a set threshold, the remaining candidates are removed, a process known as non-great value suppression. The purpose of non-great value suppression is to remove redundant candidate frames and ensure that only the best candidate frames are retained in each category. This step is repeated until all categories are traversed, ensuring that only one optimal candidate is retained for each category. After completing the non-extremely large value suppression, the remaining candidate boxes are further filtered. The remaining candidate frames in each category are then fine-tuned using multiple category-specific regressors designed to optimize their position and size. Eventually, each category will output a regression-corrected and highest-scoring edge box as the final detection result of the target in that category. As for the basketball players during the motion state process, modules with tracking functions are also added to the detection process because the people are constantly moving.

### 2.3. Tracking model based on multi-feature fusion algorithm

The athletes in basketball sports scenarios have a significant degree of appearance resemblance during the multi-person monitoring procedure. Moreover, their movements on the court are frequent and staggered, and once staggered movement or body overlap occurs, it is difficult for the tracking algorithm to accurately differentiate and recognize each athlete, which leads to the frequent problem of misidentification. For this reason, a tracking model for basketball players is constructed by combining multiple features and fusing them. The tracking model is based on a simple real-time tracking algorithm, and the next position of the athlete is judged by recursive filtering, which has a better prediction effect for the situation of having people in the shade [17]. Recursive filtering by analyzing the state parameters of the target at different moments for the corresponding next moment position judgment, in the output results will also be based on the real-time state of the target to correct the results [23]. The recursive filtering calculates the state of the target at different moments is shown in Equation (10).

$$Z_{t+1} = JZ_t + K_j I_{t+1}, \quad (10)$$

where  $Z_{t+1}$  is the state of the target at moment  $t + 1$ ,  $Z_t$  is the state of the target at the moment,  $J$  is the parameter switching matrix,  $K_j$  is the manipulation matrix, and  $I_{t+1}$  is the input moment value at moment  $t + 1$ . The formula for recursive filtering to calculate the covariance moment values of the state parameters at different moments of the target after predicting the state parameters at different moments of the target is shown in Equation (11) [12].

$$X_{t+1} = JX_t \times J^T + V, \quad (11)$$

where  $X_{t+1}$  is the state parameter under moment  $t + 1$ ,  $X_t$  is the state parameter at moment  $t$ ,  $J^T$  is the moment value operation coefficients at any moment, and  $V$  represents the noise moment value. Before the recursive filtering is about to output the predicted state, the output results are also corrected according to the target state parameters recognized at the current moment. The value-added calculation of recursive filtering is shown in Equation (12).

$$G_{t+1} = \frac{\bar{X}_{t+1}C^{T_r}}{(C\bar{X}_{t+1}C^{T_r} + \bar{V})}, \quad (12)$$

where  $G_{t+1}$  represents the recursive filtering under moment  $t + 1$ ,  $T_r$  represents any moment,  $\bar{X}_{t+1}$  represents the detected real-time state parameters under moment  $t + 1$ ,  $C$  represents the detected moment value, and  $\bar{V}$  represents the detected real-time noise covariance moment value. Equation (13), which calculates the best estimate of the target's state parameters, illustrates the process.

$$Z_{t+1}^R = \bar{Z}_{t+1} + G_{t+1}(g_{t+1} - C\bar{Z}_{t+1}), \quad (13)$$

where  $Z_{t+1}^R$  represents the best estimate under moment  $t + 1$ ,  $\bar{Z}_{t+1}$  is the detected real-time parameters under moment  $t + 1$ , and  $g_{t+1}$  is the detected parameters under moment  $t + 1$ . The corrected covariance moment values of the state parameters are calculated as shown in Equation (14).

$$\bar{X}_{t+1} = (1 - G_{t+1}C)\bar{X}_{t+1}, \quad (14)$$

where  $\bar{X}_{t+1}$  represents the corrected state-parameter covariance moment values under moment  $t + 1$ . Based on  $\bar{X}_{t+1}$ , the target-parameter position prediction under  $t + 2$  can be performed. When dealing with the multi-target following task, in order to efficiently approve the targets in consecutive frames, metrics such as intersection and merger ratios or feature similarity distances are often utilized to construct a loss moment value. The construction of this moment value lays the foundation for the subsequent data association step, and the core of constructing the loss moment value lies in transforming the multi-target following task into an optimal allocation problem. To handle such allocation difficulties, the Hungarian method is applied. Its basic principle is to work on a lossy moment value with equal rows and columns. Among them, each row of the moment values represents a goal in the previous moment, while each column corresponds to a goal in the next moment. The goal of the Hungarian algorithm is to find multiple elements of loss moments with the smallest loss without violating the “one row, one column” principle. Minimum loss elements are ideally 0, which represents no loss or the best match. The row and column indices of these elements directly indicate the correct correspondence of the targets in the preceding and following frames. With the Hungarian algorithm for loss moment values, the multi-objective following problem is transformed into a problem of finding the optimal set of elements in the loss moment values for a particular

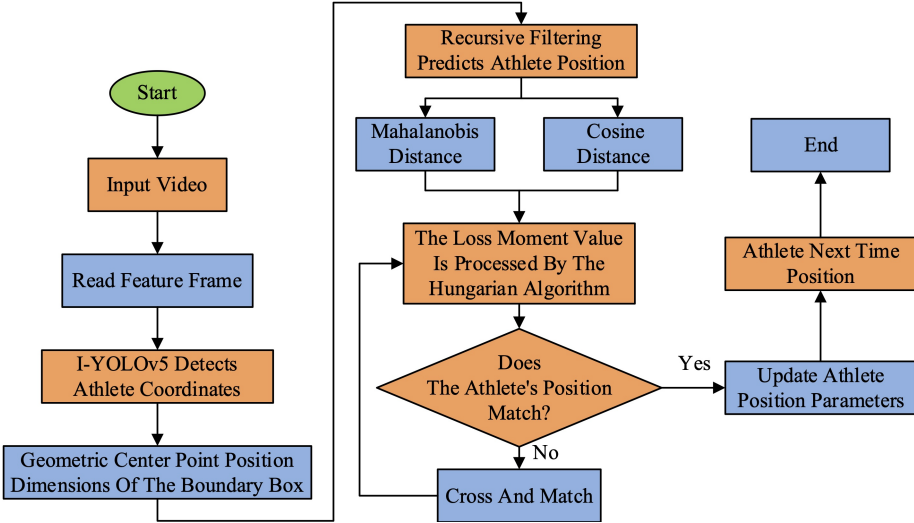


Fig. 7. Tracking model structure diagram of fusion algorithm.

pattern. These elements not only minimize the matching loss, but also ensure a one-to-one mapping between the targets in the front and back frames, resulting in an efficient and accurate TT association. An effective strategy in determining the final match is to combine the behavioral correlation with the appearance correlation, which is usually achieved by introducing a conditioning factor in the correlation evaluation model. The adjustment coefficient allows the system to flexibly adjust the weights between the two according to the actual application scenarios, thus calculating a more comprehensive and accurate athlete association. Equation (15) displays the correlation degree calculation.

$$D_{a_y, b_y} = k_t d^m(a_y, b_y) + (1 - k_t) d^y(a_y, b_y), \quad (15)$$

where  $D_{a_y, b_y}$  represents the association of the  $b_y$ th athlete on trajectory  $a_y$ ,  $k_t$  represents the moderating coefficient,  $d^m$  represents the horse distance, and  $d^y$  represents cosine distance. The final result can be obtained through the correlation degree, which is combined with the detection network to form the tracking model of the fusion algorithm to track the state of the basketball player at different moments. The whole flowchart is shown in Figure 7.

In the basketball sports video tracking model, the real-time position data of the athlete is initially extracted by the video detector. This includes the geometric center point position, the size of the BOB, and further extends to include the parameters of the velocity component. This comprehensive approach allows for a detailed portrayal of the athlete's motion state. Subsequently, recursion is used to predict the future position of

the athlete and combined with the features extracted from the athlete behavior detection network to enhance the robustness of tracking. Next, the system constructs a comprehensive correlation matrix for evaluating the similarity between the detection and the existing trajectory by calculating the cosine similarity of the appearance features and making a prediction of the position. The relation between the tracked object and the detection is swiftly ascertained by using an efficient Hungarian algorithm to the problem of best matching of similar moment values. After a successful match, the tracking frame is directly output and the trajectory parameters are updated. For a failed match, the system tries to perform a secondary correlation by calculating the intersection and merger ratios to capture possible missed matches. For long time unsuccessful matching trajectories, the system will clean up to avoid resource waste. Meanwhile, the newly appeared unmatched detections are regarded as the starting point of the initial vectors to initiate the tracking. The whole process continues to iterate until all frames of the video are processed. In each iteration, the system dynamically adjusts the tracking strategy based on the latest information to ensure accurate tracking of athletes in complex sports scenes.

### 3. Results

#### 3.1. Algorithm performance comparison

To ensure the efficiency of the tracking model, the operating system used for the experimental study is Windows 10, CPU is Intel Core i9-13900K @ 5.80 GHz, GPU is GeForce RTX 3070Ti, RAM is 32 G, programming language is Python 3.8, and the development environment is PyTorch 1.5. The datasets used for the experimental training and validation process are Detectron dataset [6, 28] and SportsMOT dataset [2, 3]. The Detectron dataset supports multiple object detection algorithms, making it suitable for diverse algorithm testing and comparison. The SportsMOT dataset focuses on multi-target tracking in sports scenes, including sports such as basketball, soccer, and volleyball. It has two characteristics: fast and variable speed movement, as well as similar but distinguishable appearance, making it suitable for evaluating the performance of algorithms in complex sports scenes. The evaluation criteria used are AR, homing rate (HR), trace operation time (TOT) and frames per second (FPS). AR is mainly to judge the accuracy of the algorithm to detect the target, the higher means the more accurate. HR is mainly to judge the performance of the model's classifier, the higher it is the better the classification effect. TOT is to judge the algorithm's computing speed, the faster the better. FPS is to judge the rate at which the algorithm handles video tracking and localization, the higher the better. The comparison algorithms are the traditional YOLOv5 algorithm and simple online and realtime tracking (SORT). Figure 8 displays the LF decrease of each algorithm during the dataset's training procedure. On the Detectron dataset, the LF of the I-YLOLv5 algorithm stabilizes when the iterations reaches about 40,000,

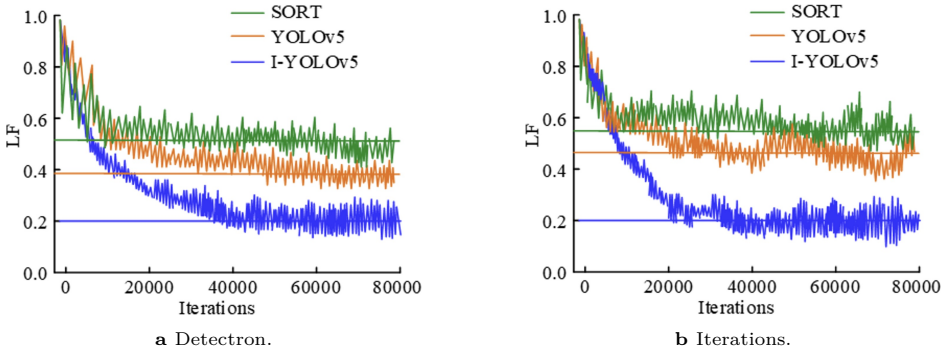


Fig. 8. Loss functions of various algorithms.

Tab. 1. Algorithm performance under different cutting ratio of image.

Model	SORT			YOLOv5			I-YOLOv5		
	AR (%)	HR (%)	TOT (s)	AR (%)	HR (%)	TOT (s)	AR (%)	HR (%)	TOT (s)
5 × 5	85.14	89.15	1.85	87.08	92.15	1.33	91.25	94.65	0.89
4 × 4	85.01	89.09	1.54	87.01	92.04	1.01	91.21	94.59	0.58
3 × 3	84.89	89.01	1.28	88.89	91.95	0.78	91.16	94.54	0.41
2 × 2	84.77	88.92	1.05	88.77	91.89	0.65	91.10	94.51	0.33
1 × 1	84.61	88.85	0.68	88.69	91.88	0.44	91.02	94.47	0.21

and the LF is 0.20. The LF of the YOLOv5 algorithm stabilizes when the iterations reaches about 50 000, and the LF is 0.38. The LF of the SORT algorithm stabilizes when the iterations reaches about 40 000 and the LF is 0.51. In Figure 8b, the SORT algorithm and the YOLOv5 algorithm have difficulty in reaching a more stable condition on the SportsMOT dataset, and the LF increases. Since the I-YOLOv5 algorithm adds multi-AM, the LF of the I-YOLOv5 algorithm can be stabilized quickly. Moreover, it can maintain around 0.2 in different datasets and the value of LF is the lowest among all the algorithms. By segmenting the image to different degrees, the detection of the segmented image by each algorithm is shown in Table 1. In this Table, the more the number of chunks of the image cut, the higher the AR and HR. As the chunks of the image becomes more, the running time of the algorithms becomes longer. After image cutting, the algorithm running time is mainly spent on the image merging process, while the coordinate AM of the I-YOLOv5 algorithm has the function of numbering each part of the image. This makes the I-YOLOv5 algorithm less affected by the image merging process, and the operation time is always kept within 1s. In the case where the image

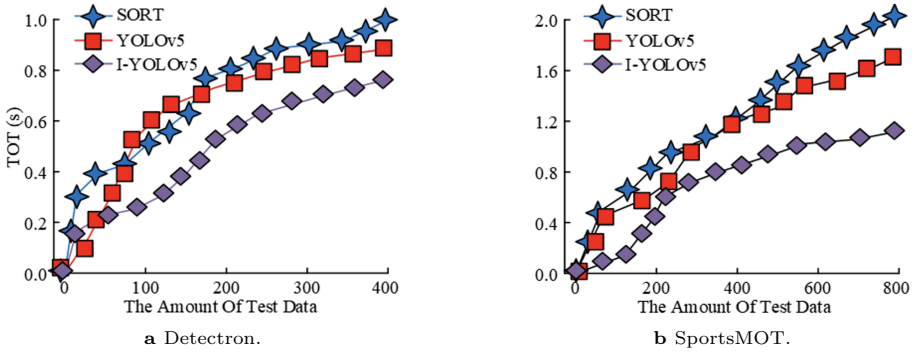


Fig. 9. Comparison of operation time of various algorithms.

is cut into  $5 \times 5$ , the running time of the I-YOLOv5 algorithm is 33.08% shorter than that of the traditional YOLOv5 algorithm. The SORT algorithm, on the other hand, has the worst performance situation, with the computation time directly exceeding 1s once the image has been cut. From this, it can be seen that the I-YOLOv5 algorithm, with its coordinate attention module, effectively reduces the impact of image merging on runtime. Even in scenes with many image cuts, the I-YOLOv5 algorithm still exhibits superior computational efficiency. The running time for each algorithm to complete the tracking on the Detectron dataset and the SportsMOT dataset is shown in Figure 9. In this Figure, the TT runtime of each algorithm on the Detectron dataset basically maintains a linear increase. Among them, the I-YOLOv5 algorithm has the shortest runtime, which is 21.26% lower than the second YOLOv5 algorithm runtime on average. In Figure 9b, the average running time of the I-YOLOv5 algorithm becomes significantly shorter when the amount of test data reaches 210 and no longer maintains the previous growth rate. This is because the TSTD module in the I-YOLOv5 algorithm makes it run faster during the training process, while the YOLOv5 algorithm and the SORT algorithm still maintain the same operation rate. From this, it can be seen that the I-YOLOv5 algorithm can stably track targets and maintain good stability even when facing large amounts of data. The algorithms are recognizing each frame of the video as an image while tracking the basketball players in the video data. The number of images per second that can be recognized by each algorithm is shown in Figure 10.

In Figure 10a, the FPS of the I-YOLOv5 algorithm on the validation dataset stays in a relatively stable state with an average value of 40, which is a 31.65% improvement over the traditional YOLOv5 algorithm. The SORT algorithm, on the other hand, has a worse performance situation, and SORT shows unstable FPS when processing some video clips with more complex personnel. In Figure 10b, affected by recursive filtering, the I-YOLOv5 algorithm got some learning during the testing process, with the ability to

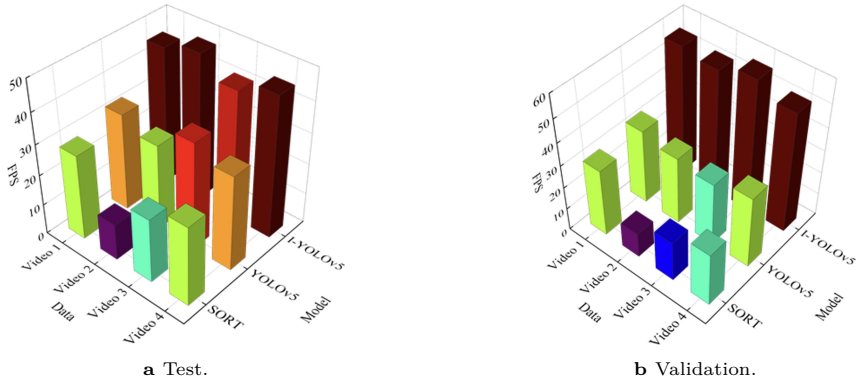


Fig. 10. Various algorithms can recognize the number of images per second.

Tab. 2. Analysis results of target tracking performance of each algorithm.

Data set	Model	Image			Video		
		SORT	YOLOv5	I-YOLOv5	SORT	YOLOv5	I-YOLOv5
Detectron	Avg-AR [%]	85.41	92.51	98.98	78.54	88.15	97.21
	Avg-HR [%]	86.14	90.89	97.87	73.84	89.18	96.25
SportsMOT	Avg-AR [%]	84.21	91.58	99.01	76.58	87.51	98.65
	Avg-HR [%]	83.15	90.67	98.59	74.35	87.68	97.21

predict the next frame. The FPS of I-YOLOv5 algorithm has been improved somewhat during the validation process, with an average FPS of 49, which is 22.50% higher than the validation process.

### 3.2. Analysis of the effect of target tracking

In terms of performance, the I-YOLOv5 algorithm has been reflected in the comparison process in the previous section, while the specific tracking effect is mainly judged by AR and HR. In order to more accurately analyze the TT effect of basketball players for the three algorithms of SORT, YOLOv5 and I-YOLOv5, average accuracy ratio (Avg-AR) and average homing rate (Avg-HR) are used for comparison. Table 2 displays the outcomes of the comparison. The I-YOLOv5 algorithm has the highest Avg-AR and Avg-HR among all the algorithms both in image target detection and video TT process. During video TT on the SportsMOT dataset, the I-YOLOv5 algorithm has an Avg-AR of 98.65% and an Avg-HR of 97.21%. Due to the fact that video has more complexity than image, the algorithms have smaller AR and HR for tracking video targets than image





Fig. 11. Target tracking and recognition effect.

target detection. Whereas the I-YOLOv5 algorithm has a recursive filter prediction module, it still maintains high AR and HR during tracking video targets.

To show the tracking situation more intuitively, the target recognition effect is demonstrated. The recognition situation is shown in Figure 11. In this Figure, in the scenario facing personnel stacking, both SORT and YOLOv5 algorithms perform poorly with missed detection. The I-YOLOv5 algorithm, on the other hand, has precise localization of the personnel position due to the coordinate AM and avoids leakage detection due to personnel stacking. In Figure 11b, the SORT algorithm incorrectly treats the off-site personnel as the detection target. However, the I-YOLOv5 algorithm has more considerations for the correlation matching of the detection targets, so as to achieve the effect of detecting the targets accurately. For the TT situation of the video, a basketball player of a basketball game video clip is used as the tracking object, and the algorithm detects the number of people in each frame of the image to visualize the situation. The video is 30 FPS, 30 seconds in total, and the actual number of basketball players is 10. To simplify the data, the average value of target detection every 5 seconds is shown.

Table 3 displays the TT outcomes for each algorithm. The I-YOLOv5 algorithm maintains a more stable state during the TT process of the video. The number of target detections for each algorithm in the 11–15 seconds segment of the video is less than the actual number of athletes because some segments of the athletes during the game are out of the video range. The YOLOv5 algorithm appears to be unable to track the target

Tab. 3. Target tracking results of each algorithm.

Time [s]	Track effect evaluation index	SORT	YOLOv5	I-YOLOv5
0–5	Target detection average	7	9	10
	Missed average	3	1	0
6–10	Target detection average	7	9	10
	Missed average	1	1	0
11–15	Target detection average	6	7	8
	Missed average	2	1	0
16–20	Target detection average	5	7	10
	Missed average	5	3	0
21–25	Target detection average	8	9	10
	Missed average	2	1	0
26–30	Target detection average	9	10	10
	Missed average	1	0	0

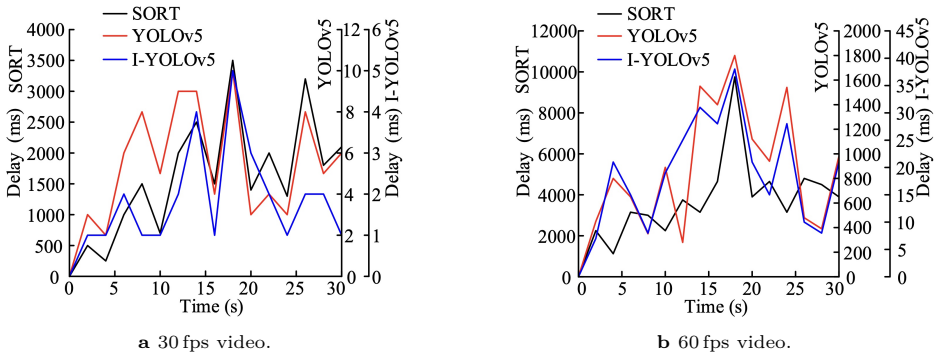


Fig. 12. The delay of each algorithm in tracking different targets.

in the 16–20 seconds segment when some of the basketball players are moving faster. At this time, the fluctuation is more obvious, and there is only a complete number of targets detected by the I-YOLOv5 algorithm. From this, it can be seen that the I-YOLOv5 algorithm can stably track targets without any missed detections, and can fully monitor all basketball players. Ordinary online game videos of basketball tend to be in 30 or 60 FPS system. By using different algorithms for TT of the video, whether or not lagging occurs is an important indicator for judging whether the algorithms can perform online tracking. The delay of each algorithm in tracking different targets is shown in Figure 12. In this figure, both the YOLOv5 algorithm and the I-YOLOv5

Tab. 4. Application test results of I-YOLOv5 and Deep-EIoU in basketball match

Index	I-YOLOv5	Deep-EIoU
Average Delay [ms]	33	67
Avg-AR	99.08	97.54
Avg-HR	99.12	96.83
CPU usage [%]	26.54	35.67
Target loss	No	No

algorithm show a better steady state during video TT at 30 frames. The TT delays are all under 10 ms, while the SORT algorithm shows a delay of up to 3500 ms. In Figure 12b, under 60 frames video, the traditional YOLOv5 algorithm showed lagging phenomenon and appeared up to 1800 ms delay. However, the recursive filtering makes the I-YOLOv5 algorithm still maintain good stability during the video TT at 60 frames, with a delay of up to 45 ms. Therefore, the I-YOLOv5 algorithm can perfectly support online real-time tracking of basketball sports videos. To further analyze the performance of the I-YOLOv5 algorithm, the study also conducted a test comparison between the I-YOLOv5 algorithm and the Deep Expansion IoU (Deep IoU) algorithm in a practical application of a basketball game.

The test results are shown in Table 4. I-YOLOv5 performs better than Deep EIoU in basketball games. The average latency of I-YOLOv5 is only 33 ms, much lower than Deep EIoU's 67 ms, demonstrating faster response capability. In terms of accuracy and regression rate, I-YOLOv5 also leads Deep EIoU with scores of 99.08% and 99.12%, respectively, surpassing Deep EIoU's 97.54% and 96.83%, indicating higher tracking accuracy. Meanwhile, the CPU usage of I-YOLOv5 is relatively low at 26.54%, which is more energy-efficient than Deep EIoU's 35.67%. Both did not experience target loss, ensuring the stability of tracking. It can be seen that I-YOLOv5 performs better than Deep EIoU in terms of speed, accuracy, and resource utilization.

#### 4. Conclusion

This research focuses on the tracking of athletes during basketball games. To ensure real-time tracking of the target, the YOLOv5 algorithm was improved by fusing the multi-feature detection module to form a new I-YOLOv5 algorithm. The image to be detected was first cut to some extent to remove redundant information. Subsequently, the target was recognized according to the feature parameters, followed by the prediction of the target's position in the next frame by calculating the cosine similarity. Finally, the prediction results were corrected by real-time images and the tracking results were output. The outcomes revealed that the I-YOLOv5 algorithm had a good performance.

The LF stabilized to 0.20 when the number of iterations reached about 40 000, and the images that could be processed per second was 49 on average. The target detection time of the I-YOLOv5 algorithm was 33.08% shorter than that of the conventional YOLOv5 algorithm when the image was cut to  $5 \times 5$ . The TT runtime of the I-YOLOv5 algorithm on the Detectron dataset was reduced by 21.26% compared to the traditional YOLOv5 algorithm. On the SportsMOT dataset, the I-YOLOv5 algorithm achieved an average accuracy of 98.65% and Avg-HR of 97.21%. The tracking latency of the I-YOLOv5 algorithm on 60 fps basketball sports videos was consistently maintained within 40 ms. In conclusion, the I-YOLOv5 algorithm exhibits a relatively short processing time and high accuracy. The I-YOLOv5 algorithm is capable of tracking the basketball player's target in real time on online videos and exhibits enhanced recognition of overlapping multiple targets. Furthermore, it is adaptable to the TT of a diverse range of basketball sports images or videos. While this research addresses the issue of tracking the movements of a basketball player, it does not extend to other types of targets. As such, additional studies are needed to examine this approach's effectiveness in various TT circumstances.

## 5. Authors' declarations

### 5.1. Conflict of interest

The authors have no conflict of interest to report.

### 5.2. Data availability

The information on the source of data is included in the manuscript.

## References

- [1] G. Bharathi and G. Anandharaj. A conceptual real-time deep learning approach for object detection, tracking and monitoring social distance using Yolov5. *Indian Journal of Science and Technology*, 15(47):2628–2638, 2022. doi:10.17485/IJST/v15i47.1880.
- [2] Y. Cui, C. Zeng, X. Zhao, Y. Yang, G. Wu, et al. SportsMOT: A large multi-object tracking dataset in multiple sports scenes. In: *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 9887–9897. IEEE Computer Society, 2023. doi:10.1109/ICCV51070.2023.00910.
- [3] Y. Cui, C. Zeng, X. Zhao, Y. Yang, G. Wu, et al. Sportsmot. GitHub, 2024. <https://github.com/MCG-NJU/SportsMOT>.
- [4] P. T. Esteves, J. Arede, B. Travassos, and M. Dicks. Gaze and shoot: Examining the effects of player height and attacker-defender interpersonal distances on gaze behavior and shooting accuracy of elite basketball players. *Revista de Psicologia del Deporte*, 30(3):1–8, 2021. <https://rpd-online.com/article-view/?id=466>.
- [5] T. Facchinetti, R. Metulini, and P. Zuccolotto. Filtering active moments in basketball games using data from players tracking systems. *Annals of Operations Research*, 325:521–538, 2023. doi:10.1007/s10479-021-04391-8.

- [6] R. Girshick, I. Radosavovic, G. Gkioxari, P. Dollár, and K. He. Detectron. GitHub, 2018. <https://github.com/facebookresearch/detectron>.
- [7] C. Guo, M. Cai, N. Ying, H. Chen, J. Zhang, et al. ANMS: Attention-based non-maximum suppression. *Multimedia Tools and Applications*, 81(8):11205–11219, 2022. doi:10.1007/s11042-022-12142-5.
- [8] M.-H. Guo, T.-X. Xu, J.-J. Liu, Z.-N. Liu, P.-T. Jiang, et al. Attention mechanisms in computer vision: A survey. *Computational Visual Media*, 8(3):331–368, 2022. doi:10.1007/s41095-022-0271-y.
- [9] Z. Hao, X. Wang, and S. Zheng. Recognition of basketball players’ action detection based on visual image and Harris corner extraction algorithm. *Journal of Intelligent and Fuzzy Systems*, 40(4):7589–7599, 2021. doi:10.3233/JIFS-189579.
- [10] M. Hasanvand, M. Nooshyar, Moharamkhani, and A. Selyari. Machine learning methodology for identifying vehicles using image processing. *Artificial Intelligence and Applications*, 1(3):170–178, 2023. doi:10.47852/bonviewAIA3202833.
- [11] L. He, J. C. W. Chan, and Z. Wang. Automatic depression recognition using CNN with attention mechanism from videos. *Neurocomputing*, 422(1):165–175, 2021. doi:10.1016/j.neucom.2020.10.015.
- [12] Y. Ji, Z. Kang, and C. Zhang. Two-stage gradient-based recursive estimation for nonlinear models by using the data filtering. *International Journal of Control, Automation, and Systems*, 19(8):2706–2715, 2021. doi:10.1007/s12555-019-1060-y.
- [13] M. Jin, H. Li, and Z. Xia. Hybrid attention network and center-guided non-maximum suppression for occluded face detection. *Multimedia Tools and Applications*, 82(10):15143–15170, 2023. doi:10.1007/s11042-022-13999-2.
- [14] Z. Li, F. Liu, W. Yang, S. Peng, and J. Zhou. A survey of convolutional neural networks: Analysis, applications, and prospects. *IEEE Transactions on Neural Networks and Learning Systems*, 33(12):6999–7019, 2021. doi:10.1109/TNNLS.2021.3084827.
- [15] Y. Liu, L. Geng, W. Zhang, and Y. Gong. Survey of video-based small target detection. *Journal of Image and Graphics*, 9(4):122–134, 2021. doi:10.18178/joig.9.4.122-134.
- [16] Y. Ma, N. Li, Zhang, S. Wang, and H. Ma. Image encryption scheme based on alternate quantum walks and discrete cosine transform. *Optics Express*, 29(18):28338–28351, 2021. doi:10.1364/OE.431945.
- [17] J. Mao, Y. Sun, X. Yi, H. Liu, and D. Ding. Recursive filtering of networked nonlinear systems: a survey. *International Journal of Systems Science*, 52(6):1110–1128, 2021. doi:10.1080/00207721.2020.1868615.
- [18] Z. Niu, G. Zhong, and H. Yu. A review on the attention mechanism of deep learning. *Neurocomputing*, 452(1):48–62, 2021. doi:10.1016/j.neucom.2021.03.091.
- [19] J. Ren and Y. Wang. Overview of object detection algorithms using convolutional neural networks. *Journal of Computer Communications*, 10(1):115–132, 2022. doi:10.4236/jcc.2022.101006. <https://www.scirp.org/journal/paperinformation?paperid=115011>.
- [20] A. Rizaldy, P. Ghamisi, and R. Gloaguen. Channel attention module for segmentation of 3d hyperspectral point clouds in geological applications. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 48:103–109, 2024. doi:10.5194/isprs-archives-XLVIII-2-W11-2024-103-2024.
- [21] H. Song, X. Zhang, J. Song, and J. Zhao. Detection and tracking of safety helmet based on DeepSort and YOLOv5. *Multimedia Tools and Applications*, 82(7):10781–10794, 2023. doi:10.1007/s11042-022-13305-0.
- [22] R. Sun, J. Kuang, Y. Ding, J. Long, Y. Hu, et al. High-efficiency differential single-pixel imaging

- based on discrete cosine transform. *IEEE Photonics Technology Letters*, 35(17):955–958, 2023. doi:10.1109/LPT.2023.3286105.
- [23] H. Tan, B. Shen, and H. Shu. Robust recursive filtering for stochastic systems with time-correlated fading channels. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 52(5):3102–3112, 2021. doi:10.1109/TSMC.2021.3062848.
- [24] Z. Terner and A. Franks. Modeling player and team performance in basketball. *Annual Review of Statistics and Its Applications*, 8(1):1–23, 2021. doi:10.1146/annurev-statistics-040720-015536.
- [25] T. Wang and C. Shi. Basketball motion video target tracking algorithm based on improved gray neural network. *Neural Computing and Applications*, 35(6):4267–4282, 2023. doi:10.1007/s00521-022-07026-6.
- [26] W. Wang, S. Wang, Y. Li, and Y. Jin. Adaptive multi-scale dual attention network for semantic segmentation. *Neurocomputing*, 460(1):39–49, 2021. doi:10.1016/j.neucom.2021.06.068.
- [27] Y. Wu, D. Deng, X. Xie, M. He, J. Xu, et al. Obtracker: Visual analytics of off-ball movements in basketball. *IEEE Transactions on Visualization and Computer Graphics*, 29(1):929–939, 2022. doi:10.1109/TVCG.2022.3209373.
- [28] Y. Wu, A. Kirillov, F. Massa, W.-Y. Lo, and R. Girshick. Detectron2. GitHub, 2019. <https://github.com/facebookresearch/detectron2>.
- [29] J. Xu, B. Ai, W. Chen, A. Yang, P. Sun, et al. Wireless image transmission using deep source channel coding with attention modules. *IEEE Transactions on Circuits and Systems for Video Technology*, 32(4):2315–2328, 2021. doi:10.1109/TCSVT.2021.3082521.
- [30] X. Yang, Y. Luo, M. Li, Z. Yang, C. Sun, et al. Recognizing pests in field-based images by combining spatial and channel attention mechanism. *IEEE Access*, 9(1):162448–162458, 2021. doi:10.1109/ACCESS.2021.3132486.
- [31] M. C. Yesilli, J. Chen, F. A. Khasawneh, and Y. Guo. Automated surface texture analysis via discrete cosine transform and discrete wavelet transform. *Precision Engineering*, 77(1):141–152, 2022. doi:10.1016/j.precisioneng.2022.05.006.
- [32] W. Zhan, C. Sun, M. Wang, J. She, Y. Zhang, et al. An improved Yolov5 real-time detection method for small objects captured by UAV. *Soft Computing*, 26(1):361–373, 2022. doi:10.1007/s00500-021-06407-8.
- [33] G. Zhaoxin, L. Han, Z. Zhijiang, and P. Libo. Design a robot system for tomato picking based on yolo v5. *IFAC-PapersOnLine*, 55(3):166–171, 2022. doi:10.1016/j.ifacol.2022.05.029.
- [34] X. Zhu, K. Guo, S. Ren, B. Hu, M. Hu, et al. Lightweight image super-resolution with expectation-maximization attention mechanism. *IEEE Transactions on Circuits and Systems for Video Technology*, 32(3):1273–1284, 2022. doi:10.1109/TCSVT.2021.3078436.