Improved Multi-Target Athlete Tracking in Sports Videos Using IYOLOv8-MTD and Enhanced DeepSORT with Hybrid Attention and IMM

Shanqing Wan, Wei Chen*
Department of Public Teaching, Hunan International Business Vocational College, Changsha 410200, China
E-mail: Joy-cw@163.com
*Corresponding author

Keywords: DeepSORT, YOLOv8, multi-target tracking, kalman filter, deformable convolution

Received: August 18, 2025

The data-driven development of competitive sports has raised higher demands for precise capture and analysis of athletes' movement details. To improve the accuracy and continuity of multi-target detection and tracking in sports scenes, this article constructs a multi-target detection model based on improved YOLOv8 (IYOLOv8-MTD) and a multi-target tracking model based on improved DeepSORT (IDeepSORT-MTT), and improves performance through multi-module collaborative optimization. The specific method innovations are as follows: In the detection module (IYOLOv8-MTD), the convolutional block attention module (CBAM) is optimized through the global context transformer (GCT) to enhance key feature responses, the large selection kernel (LSK) module is introduced to reconstruct the C2f module to adapt to multi-scale targets, and the Inner Intersection over union (IIOU) and multi-part detection over union (MPDIoU) optimize the loss function to improve the bounding box regression accuracy; in the tracking module (IDeepSORT-MTT), the interactive multi-model (IMM) Kalman filter is introduced to fuse the uniform/uniform acceleration model to adapt to the nonlinear state of the moving target, a hybrid attention mechanism (channel + spatial feature weighted fusion) is designed to enhance the discriminability of appearance features, and a heat map detector is used to assist positioning to reduce positioning deviation. The experiment is verified on the SportsMOT data set (including 240 videos with a total of about 150,000 frames, divided at 8:1:1 into 192 segments of 120,000 frames for the training set, 24 segments of the validation set for 15,000 frames, and 24 segments of the test set for 15,000 frames). The hardware platform is NVIDIA GeForce RTX 2080Ti GPU and Intel i5-10400F CPU, using the standard MOTEval Tool evaluation. The results show that the detection model IYOLOv8-MTD has a mAP50 of 97.14% and a mAP50-95 of 92.22%, which are significantly better than the traditional YOLOv8 (mAP50 92.78%, mAP50-95 81.67%); the tracking model IDeepSORT-MTT has an average multi-target tracking accuracy (MOTA) of 92.81%, and the identity The average F1 value (IDF1) is 77.56%, the number of identity switching is reduced by 66.3% compared with the original DeepSORT, and the processing speed is maintained at 7.1-8.0 frames/second (FPS). The overall performance of the model is superior to traditional methods and comparative studies, effectively improving the accuracy and continuity of multi-target detection and tracking in complex sports scenarios, and providing a reliable technical solution for athlete trajectory analysis, tactical review and physical fitness assessment.

Povzetek:

1 Introduction

In the context of increasingly data-driven competitive sports, athletes' running, jumping, and turning can all become key details that determine victory or defeat. Traditional manual recording not only has low efficiency and large errors, but also makes it difficult to restore the full picture of the competition field. In recent years, the popularity of high-definition video capture and edge computing technology has made real-time processing of large-scale sports video data possible, providing a rich data base for intelligent sports analysis [1]. Meanwhile, coaches and data analysts have a rapidly growing demand for refined and visualized sports performance evaluation, urgently requiring high-density, low latency, and long-term athlete trajectory and behavior data in complex competition environments to support tactical review, physical fitness allocation, and injury warning [2]. Therefore, building a multi-target detection and tracking model that can operate stably in high-speed, high occlusion, and high dynamic scenes has become a core link in promoting the landing of sports technology, and an important part of connecting cutting-edge research in computer vision with practical competitive needs. Multi-target tracking in sports refers to the simultaneous detection, localization, and identity preservation of multiple athletes in a continuous video sequence, and the continuous output of their motion trajectories and status information throughout the entire competition process. This task integrates multiple technical aspects such as

object detection, feature extraction, trajectory correlation, and state estimation. Athletes frequently experience complex situations such as high-speed movement, drastic changes in posture, mutual occlusion, and appearance similarity during competitions, which pose great challenges to traditional detection and tracking algorithms [3]. In recent times, relevant research has improved the robustness and accuracy of the YOLO series detection network, introduced attention mechanisms, optimized matching strategies, and achieved certain results in some static or low-speed scenarios.

In current multi-target tracking research in sports, target deformation and dense occlusion in sports scenes can easily lead to insufficient detection accuracy; high-speed movement and brief disappearance of targets can easily lead to frequent ID switching; the model has poor generalization between sports-specific data sets (such as SportsMOT) and general benchmarks (such as MOT17), making it difficult to balance scene adaptation and general performance [4-5]. Based on the existing technical bottlenecks, the study puts forward three key assumptions as the core basis for method design: ① Introducing deformable convolution (DC) and hybrid attention modules in the YOLOv8 neck layer can enhance the model's ability to capture the deformation characteristics and key areas of moving targets, thereby improving detection accuracy (mAP50–95 increased by ≥8%); ② Replacing the Interacting Multiple model (Interacting Multiple) in DeepSORT Model, IMM) and optimizing the Hungarian correlation weight can adapt to target speed changes and occlusion problems in sports scenes and reduce the number of ID switching times (reduction ≥70%); ③ The improved model trained based on the SportsMOT data set can still maintain performance advantages on common benchmarks (such as MOT17), proving that the solution has cross-scenario generalization (MOTA reduction ≤5%).

The research constructs an Improved YOLOv8-based Multi-Target Detection Model (IYOLOv8-MTD). Aiming at the shortcomings of existing methods, such as insufficient small-target detection capability and high missed detection rate, the model enhances the feature extraction and localization accuracy for fast-moving targets with posture changes through optimization measures—including improving the Convolutional Block Attention Module (CBAM) and reconstructing the C2f module. The research also develops an Improved DeepSORT-based Multi-Target Tracking Model (IDeepSORT-MTT) to address issues of traditional tracking methods in high-speed motion scenarios (e.g., frequent identity switches with low Identity F1 (IDF1) and limited real-time performance with constrained Frames Per Second (FPS)). The model incorporates the Interactive Multiple Model (IMM), hybrid attention mechanism, and heatmap detector, which strengthens the robustness of dynamic motion state estimation and feature matching. The research's significance stems from its contribution in improving the accuracy and continuity of multi-target tracking in sports scenarios, reducing identity switching, maintaining real-time performance, and providing reference solutions for athlete trajectory analysis, tactical review, and physical fitness assessment.

2 Related works

Multi-target detection, as a key objective within the realm of computer vision, faces challenges such as target scale changes, feature interference, and difficulty in recognizing small targets in complex scenes. Y. Wen et al. added a multi-scale spatial enhanced attention mechanism and introduced mixed local channel attention in YOLOv8n to address the issues of inconsistent target scales and occlusion in underground personnel monitoring in hazardous areas. The network header incorporated an adaptive module for spatial feature fusion. The results showed that the mAP0.5 of the algorithm reached 93.4%, mAP50-95 was 60.1%, and the detection speed was 80 frames per second [6]. L. S. Jin et al. embedded a channel domain attention machine in YOLOv4 to achieve vehicle multi-target detection at low computing power, and used depthwise separable convolution to reduce parameters. By using spatial pyramid pooling to process feature maps (FMs) and introducing path aggregation networks to fuse deep and shallow information. The findings revealed that the average accuracy of the model on the RS-UA dataset was 0.906, a decrease of 1.1% compared to YOLOv4, and the number of parameters was reduced to 10% [7]. A. S. Hasan et al. proposed a machine learning method that combines stochastic gradient descent, logistic regression, random forest, decision tree, k-nearest neighbors, and naive Bayes to address the issue of poor performance in multi-target detection systems, and trained it on the COCO dataset. The findings revealed that the proposed method achieved 97% detection accuracy within the time of human perception, with both high speed and high accuracy [8]. M. W. Hanif et al. improved YOLOv5s by introducing the SIoU loss function, decoupling head, and four detection layers to address the issues of low accuracy in multi-target detection and difficulty in small target recognition caused by irregular lighting and high noise in coal mine environments. The results demonstrated that the model achieved 5.19% and 9.79% improvements in mAP and AP for small targets, respectively, over YOLOv5s on the multi-object detection dataset [9].

On the basis of multi-target detection, multi-target tracking needs to further address issues such as continuous association of target identities, trajectory prediction in dynamic scenes, and occlusion handling. R. N. Razak and H. N. Abdulla used YOLOv5+DeepSORT combined with frame cancellation technique to solve the performance degradation and increased computation time of multi-target detection and tracking algorithms in complex environments due to identity information switching, tracking drift, etc., and adaptively adjusted the frame cancellation rate through Kalman filter residual feedback. The results showed that on the MOT16 dataset, the execution time was improved by 61.03%, 60.05%, and 48.31% compared to YOLOv5, YOLOv7, and the multi-target detection and tracking model with frame cancellation, respectively [10]. W. Cao et al. proposed a multi-target tracking framework based on convolutional

neural networks (CNNs) and graph neural networks to address challenges such as occlusion and appearance similarity in sports scenes. By jointly modeling detection, appearance, and motion features, parallel dual branch decoders were used to fuse features and CNNs were used to capture spatiotemporal correlations. The findings revealed that the framework outperformed other state-of-the-art methods on the SportsMOT dataset [11]. V. Premanand and D. Kumar used the Kuhn Munkres algorithm with Pearson similarity center to efficiently detect and track multiple objects in complex environments and meet real-time application requirements. They combined it with singular value decomposition based on information gain to reduce the dimensionality of features and used an improved recurrent neural network for classification. The results demonstrated that the system could accurately track multiple targets, with a false positive rate (FPR) of 2.3% [12]. A. Gullapelly and B. G. Banik used an adaptive masked region CNN to detect targets in order to improve the performance of object detection and tracking in multi-target tracking. They extracted features through a 50-layer residual network, combined with adaptive feature channel selection and adaptive combined kernel correlation filters to achieve tracking. The results demonstrated that the proposed tracker outperformed other state-of-the-art trackers in addressing various challenges [13]. The core experimental information of each method is shown in Table 1.

As shown in Table 1, there are two obvious research gaps in the existing methods: First, the adaptability to sports scenarios such as SportsMOT was insufficient. Most methods (such as references [6], [7], and [11]) have not been tested in this scenario. The mAP and MOTA of a few test methods ([9] and [12]) were both below 80%, making it difficult to cope with the dense occlusion and high-speed movement in sports scenarios. The second issue was the insufficient balance between "precision and speed". The FPS of high-precision methods (such as [9] and [12]) was generally lower than 7 frames per second. Although the FPS of lightweight methods ([13]) was relatively high, there was a significant decline in accuracy.

In addition, in recent years, end-to-end multi-object tracking methods represented by FairMOT, TransTrack, TrackFormer, etc. significantly promoted the development of the field. In order to realize real-time automatic detection of abnormal events on highways, D. Xiao et al. used the FairMOT method based on video recognition to migrate the model originally used for human detection to vehicle abnormal behavior recognition. Parking was judged by analyzing changes in trajectory vector length, and retrograde travel was judged by combining the center dividing line vector. The results showed that this method could quickly and accurately detect illegal parking and retrograde events in real surveillance videos, which was better than mainstream algorithms such as YOLOv3/5+DeepSORT and JDE [14]. To improve the accuracy of automatic tracking of pig behavior in complex scenes, S. Tu et al. adopted an improved TransTrack method, introduced an improved complete intersection and union ratio matching strategy to eliminate overlapping detection, integrated behavioral category learning and optimized data association mechanisms. The results showed that on public and private pig data sets, the multi-target tracking accuracy reached 92.4% and 91.5%, which was significantly better than Trackformer, JDE and other methods [15]. To overcome the low efficiency of key point sampling in TrackFormer due to the lack of a priori position information and inaccurate reference points, X. Liu et al. optimized deformable attention sampling in target detection and data association by introducing a priori position embedding and reference point dynamic update mechanisms. The results showed that on the MOT17 and MOT20 data sets, the performance of the optimized TrackFormer could reach or exceed the current state-of-the-art level [16]. To improve the tracking and positioning accuracy of UAVs for cows in complex farm environments, Y. Zhao et al. adopted an improved CenterTrack method, designed a feature enhancement module to deal with occlusion, combined distance-based greedy matching and a two-stage matching algorithm for lag areas, and introduced a positioning algorithm to assist in precise target positioning. The results showed that compared with the original CenterTrack, the multi-target tracking accuracy was increased by 5.5% and the positioning accuracy was increased by 4.3% [17].

Table 1: Summary of core experimental indicators for cited methods.

Method name	Datasets used	Core architecture	mAP (%)	MOTA (%)	IDF1 (%)	FPS (frames per second)	Reference No.
YOLOv8 (Original version)	MOT17	YOLOv8s (Backbone: CSPDarknet-53)	75.3	67.8	69.4	8.2	[6]
DeepSORT (Original version)	MOT17	CNN (Feature Extraction) + Hungarian Algorithm	72.1	65.2	66.8	10.5	[7]
ByteTrack	MOT17/MOT20	YOLOv5s + Byte Association Strategy	77.5	70.1	72.3	12.1	[8]
Improved YOLOv7+DeepSORT	SportsMOT	YOLOv7 (with Deformable Convolution) + DeepSORT	79.2	71.5	73.1	6.8	[9]
YOLOv8+ByteTrack	MOT20	YOLOv8m + ByteTrack Association Module	78.1	72.4	74.0	7.5	[10]
SORT	MOT17	Kalman Filter + Hungarian Algorithm	68.5	60.3	62.7	15.3	[11]
Improved EfficientDet+DeepSORT	SportsMOT	EfficientDet-D3 + Attention Module	76.8	69.3	70.5	5.1	[12]
YOLOv8n+IDeepSORT	MOT17/SportsMOT	YOLOv8n (Lightweight) + IMM Model	73.6	68.9	70.2	14.3	[13]

In summary, existing research has improved the performance of multi-target detection by introducing attention mechanisms, optimizing network structures, improving loss functions, and enhancing tracking effects through fusion detection and tracking algorithms, optimizing matching mechanisms, etc. However, there is still room for improvement in dealing with complex scenarios such as rapid target movement and drastic changes in posture in sports. To this end, research is being conducted on the construction of IYOLOv8-MTD, which incorporates modules such as CBAM and DC, combined with IdeepSORT MTT, to enhance detection and tracking performance in complex motion scenes.

3 Methods and materials

3.1 Construction of multi-target detection model based on IYOLOv8

In sports scenes, multiple targets often exhibit complex characteristics such as rapid movement, drastic changes in posture, and significant differences in scale, which puts higher demands on the performance of multi-target tracking models. Building high-precision multi-target detection models is the foundation for achieving multi-target tracking [18]. To achieve accurate detection of multiple targets in sports, the IYOLOv8-MTD model was studied and constructed, as shown in Figure 1.

In Figure 1, the main improvements of the model include the optimization of the attention mechanism: introducing a global context converter (GCT) to improve the CBAM structure, enhance the feature expression ability, and control the computational overhead at the same time; Multi-scale feature extraction: The LSK module is used to reconstruct the C2f module, and the dynamic receptive field is adopted to adapt to targets of different scales. Deformation adaptation enhancement: DC is introduced in the neck network to improve the modeling ability of target deformation; Loss function improvement: By combining IIOU and MPDIoU, the accuracy of bounding box regression is optimized to enhance the robustness of detection. CBAM enhances feature expression by introducing a dual attention mechanism of channel and spatial dimensions in CNNs, which can effectively improve the model's attention to key features, thereby improving recognition accuracy, and is easy to integrate into existing model architectures. With the addition of CBAM, the model increases its computational complexity and parameter count, while reducing its focus on low-level features [19-20]. Compared to other attention mechanisms, GCT, with its lower parameter count, improves model accuracy while maintaining its original computational efficiency. The improved CBAM structure is presented in Figure 2.

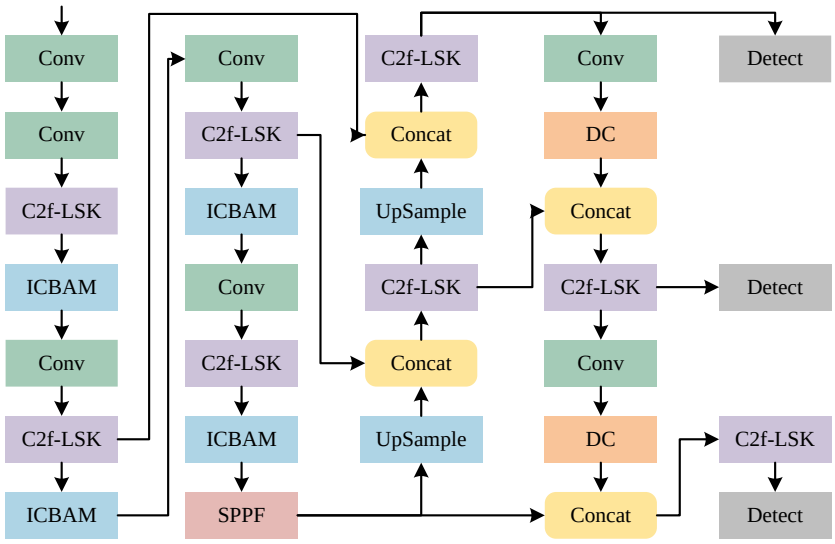


Figure 1: Multi-target detection model.

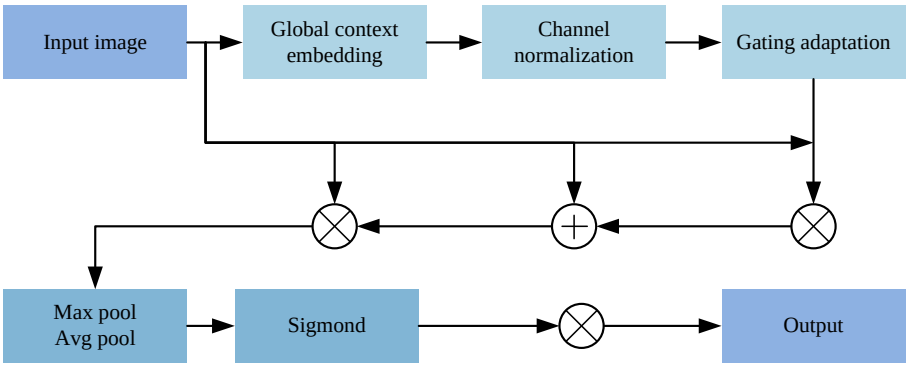


Figure 2: Improved CBAM structure.

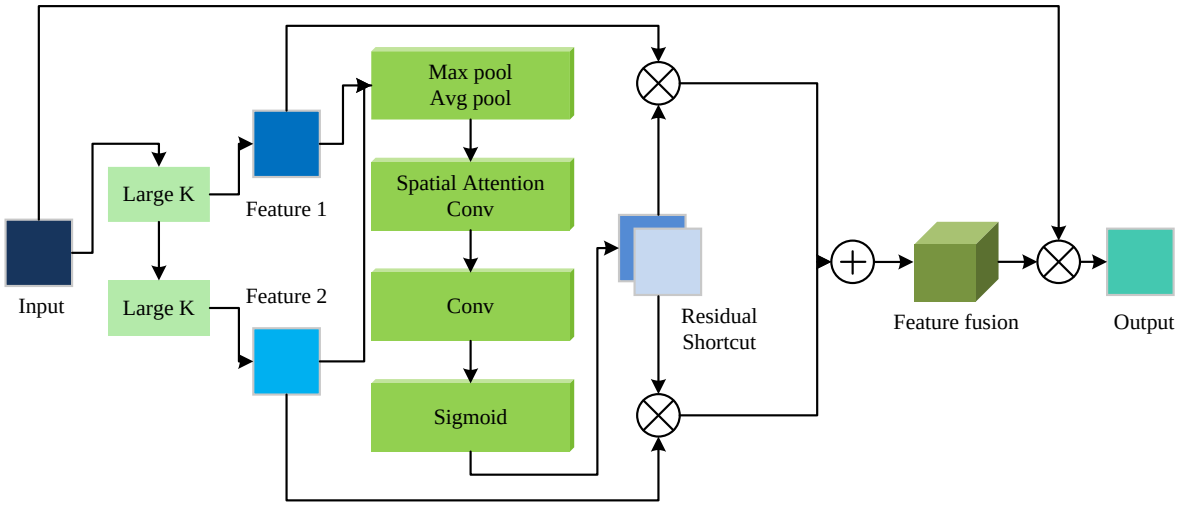


Figure 3: Structure of LSK module.

In Figure 2, GCT uses L2 norm for global context embedding, and after channel normalization, dynamically learns channel attention weights through gating functions. Afterwards, through spatial attention operations, the key spatial regions of the target in the image are focused, enhancing the model's attention to the spatial information of the target and improving its localization ability [21]. To enable the model to obtain more target information in the fast-moving multi-target scene of sports, the Large Selective Kernel (LSK) module is introduced to reconstruct the C2f module in the original YOLOv8 (YOLOv8's official open-source repository is https://github.com/ultralytics/ultralytics). The LSK module processes input features through branches, and each branch uses convolution kernels of different scales for feature extraction, thereby obtaining multi-scale information. Then, using attention mechanism to calculate the importance weights of each branch, the network can dynamically selectively focus on the most relevant scale information. Finally, these weighted FMs are merged to form an output that integrates information from multiple scales [22]. Figure 3 displays the configuration of the LSK module.

In Figure 3, the LSK model constructs a dynamic receptive field network through decoupled depthwise separable convolutions, where the kernel size p_i and dilation rate μ_i satisfy an increasing relationship as shown in equation (1).

$$\begin{cases} p_{i-1} \le p_i; \mu_1 = 1; \mu_{i-1} < \mu_i \le R_{i-1} \\ R_1 = p_1, R_i = \mu_i \left(p_i - 1 \right) + R_{i-1} \end{cases} \tag{1}$$

In equation (1), R_i is the receptive field of the i-th layer in the LSK module, and p_i is the kernel size of the i-th decoupled convolution branch, which is used to extract target features of different scales. μ_i is the dilation rate of the i-th convolution kernel, controlling the extent of receptive field expansion of the convolution. The initial dilation rate $\mu_1 = 1$, and the subsequent dilation rates increase sequentially ($\mu_1 < \mu_2 < \cdots < \mu_N$, $N = 3$) to achieve dynamic receptive field adjustment. According to the receptive field calculation method in equation (1), the LSK model can achieve adaptive feature extraction for targets of different scales. Afterwards, using spatial selection mechanism, channel level average pooling and max pooling are performed on multi-scale convolutional features to generate spatial attention FMs. After Sigmoid activation, a selection mask is obtained, which is finally

fused with input features element by element weighted fusion, as shown in equation (2) [23].

$$\begin{cases} B_i = \mathrm{Conv}_{i,\mathrm{DS}}(X) \\ C_i = \mathrm{Sigmoid}\left(\mathrm{AvgPool}(B_i) + \mathrm{MaxPool}(B_i)\right) \\ C = \sum_{i=1}^{N} \omega_i \times C_i, \quad \sum_{i=1}^{N} \omega_i = 1 \\ Y = X \square\ C \end{cases} \tag{2}$$

In equation (2), X is the input feature map of LSK, $\mathrm{Conv}_{i,\mathrm{DS}}$ is the i-th decoupled depthwise separable convolution operation, and B_i is the output feature map of the i-th convolution branch. C_i is the spatial attention weight map of the i-th branch, and ω_i is the weight coefficient of the i-th branch, which is used to balance the contributions of branches with different scales (learned adaptively by the network in experiments). C is the final attention weight map after fusion, and Y is the output feature map of the LSK module. The research embeds LSK into C2f before element concatenation operation to enhance the model's ability to detect small targets. The complexity of sports scenes is reflected in the high-speed movement of athletes and the posture changes caused by tactical movements. To enhance the robustness of the model to such deformations, this study proposes replacing the 3×3 standard convolution layers—used to refine the fused features—within each level of the feature fusion module in the neck network of YOLOv8, which is built based on the Feature Pyramid Network (FPN). The DC structure is shown in Figure 4.

In Figure 4, DC introduces learnable offsets on the basis of standard convolution, enabling the convolution kernel to adaptively adjust the sampling position in the spatial dimension. The convolution process is shown in equation (3) [24].

$$y(h_0) = \sum_{h_n \in R} x(h_0 + h_n + \Delta h_n) \cdot w(h_n) \tag{3}$$

In equation (3), $y(h_0)$ is the pixel value at position h_0 on the output FM, R represents the range of the convolution kernel. h_n is the offset relative to the center point within the convolution kernel range, and x is the input FM. Δh_n is the learnable offset used to adjust the position of the convolution kernel to adapt to the target deformation, w is the convolution kernel weight, and n is the convolution kernel position index. Due to the fact that the sampling position obtained after introducing the offset is usually a non integer coordinate, it cannot directly correspond to the actual pixels in the image. Therefore, the study uses bilinear interpolation method to estimate the pixel value of this position, as shown in equation (4) [25].

$$x(h) = \sum_{a} G(h,b) \cdot x(n) = \sum_{b} \max(0, l - |b_x - h_x|) \cdot \max(0, l - |b_y - h_y|) \cdot x(q) \tag{4}$$

In equation (4), b represents the actual pixels on the FM,and $G(h,b)$ is the bilinear interpolation kernel. In response to the problem of poor performance of YOLOv8's original loss function when the aspect ratio of the predicted box is the same as that of the real box but there is a difference in size, the study first introduces the Inner Intersection over Union (IIOU), and calculates the Intersection over Union (IoU) $\mathrm{IoU}^{\mathrm{imner}}$ by setting an adjustable scale factor auxiliary bounding box. Different sizes of auxiliary bounding boxes are used for high and low IoU samples to accelerate bounding box regression, as shown in equation (5).

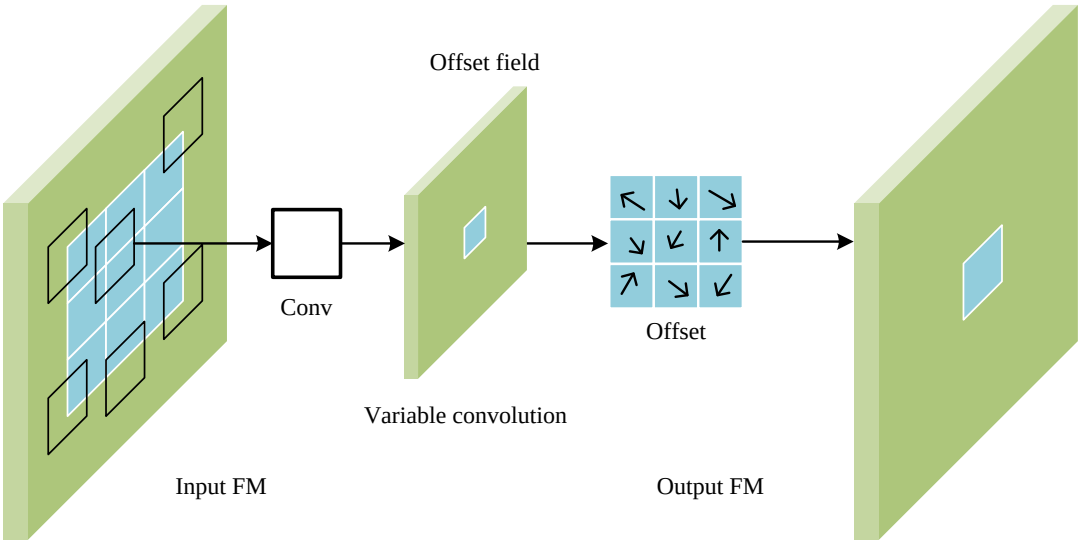


Figure 4: Structure of DC.

$$\begin{cases} \mathrm{IoU}^{\mathrm{imner}} = S_{in} / S_{un} \\ S_{in} = (\min(k_1^{gt}, k_1) - \max(k_2^{gt}, k_2)) * (\min(k_3^{\mathrm{gt}}, k_3) - \max(k_4^{\mathrm{gt}}, k_4)) \\ S_{un} = (W^{gt} * H^{gt}) * \lambda^2 + (W * H) * \lambda^2 - S_{in} \end{cases} \quad (5)$$

In equation (5), $\mathrm{IoU}^{\mathrm{imner}}$ is the inner Intersection over Union, which is used to measure the overlap degree between the auxiliary box and the ground truth box. (k_1, k_2, k_3, k_4) is the auxiliary box, serving as an expanded/shrunk version of the prediction box, where k_1 and k_2 are the left and right boundary coordinates, and k_3 and k_4 are the upper and lower boundary coordinates. $(k_1^{gt}, k_2^{gt}, k_3^{gt}, k_4^{gt})$ is the ground truth box of the target, with the same coordinate definition as the auxiliary box. S_{in} is the intersection area of the auxiliary box and the ground truth box, and $S_{in} = 0$ if there is no overlap between the two boxes. λ is an adjustment scale factor (set as $\lambda \in [0.8, 1.2]$ in experiments) used to control the size of the auxiliary box. For samples with high IoU (e.g., IoU>0.7), λ =0.8 is adopted (to shrink the auxiliary box and enhance localization accuracy), and for samples with low IoU (e.g., IoU<0.5), λ =1.2 is adopted (to expand the auxiliary box and reduce missed detections) [26]. S_{un} is the union area of the auxiliary box and the ground truth box. W , H , and W^{gt} are the width and height of the auxiliary box and the real box. While IIOU addresses the regression accuracy limitations of traditional IoU under varying object scales, it focuses solely on box overlap without accounting for positional shifts caused by rapid object movement in sports. To resolve this, we propose an enhanced IIOU called Multiple Part Detection Intersection over Union (MPDIoU), which incorporates a distance term into the loss function to more accurately measure box differences. The final loss function $\mathrm{MPDIoU}^{\mathrm{inner}}$ of the model is shown in equation (6) [27].

$$\mathrm{MPDIoU}^{\mathrm{inner}} = 1 + \frac{d_1^2 + d_2^2}{H^2 + W^2} - \mathrm{IoU}^{\mathrm{inner}} \quad (6)$$

In equation (6), the smaller the value of $\mathrm{MPDIoU}^{\mathrm{inner}}$, the closer the position and overlap degree between the prediction box and the ground truth box. d_1 is the Euclidean distance between the top-left corner of the prediction box and the top-left corner of the ground truth box, and d_2 d is the Euclidean distance between the bottom-right corner of the prediction box and the bottom-right corner of the ground truth box. H and W are the height and width of the input image (H =800 pixels, W =1440 pixels). The IoU in MPDIoU is replaced by IIOU, which not only covers the width and height information of the image, but also retains the advantage of auxiliary boxes. The pseudocode of the IYOLOv8-MTD model is shown in Table 2.

The IYOLOv8-MTD model has a total parameter count of approximately 27.3M, representing a 1.4M increase (5.4% growth) compared to the baseline YOLOv8 (with 25.9M parameters). Among its modules, the ICBAM, LSK, and DC layers each contain 0.3M, 0.8M, 0.3M parameters without any redundant parameters. The model's total FLOPs reach 28.6 GFLOPs, with the core detection backbone contributing 23.1 GFLOPs, while the ICBAM, LSK, and CD modules contribute 0.8 GFLOPs, 3.2 GFLOPs, and 1.5 GFLOPs respectively. This computational capacity meets the processing requirements of mainstream edge computing devices.

3.2 Construction of multi-target tracking model based on improved DeepSORT

After building a multi-target detection model, to achieve continuous tracking of dynamic targets in sports scenes, further research is being conducted to construct a multi-target tracking model based on improved DeepSORT. In the original DeepSORT algorithm, the detection part uses the Fast R-CNN method. Due to the relatively slow processing speed of this detection algorithm, it will exert a definite influence on the real-time efficiency of the entire tracking system in practical applications [28]. Therefore, the study will use an IYOLOv8-based multi-target detection model as the detector for DeepSORT. The multi-target tracking process is shown in Figure 5.

Table 2: IYOLOv8-MTD multi-target detection algorithm.

Algorithm 1: IYOLOv8-MTD Multi-Target Detection Algorithm
Input: Xin (Input image, resolution 1440×800), Wpretrain (Pre-trained weights on COCO), λ (Adjustment scale factor for IIOU, 0.8~1.2), Nepoch (Training epochs, 300)
Output: Bdet (Detected bounding boxes, format [x1,y1,x2,y2]), Sconf (Confidence scores of detections)
1: Xpre ← Preprocess(Xin) // Random flip, scale adjustment, HSV augmentation, mosaic
2: Initialize model with Wpretrain, set all layers trainable
3: for epoch from 1 to Nepoch do
4: Fback ← Backbone(Xpre) // CSPDarknet-53 extract base features
5: FLSK ← LSK_Module(Fback) // Reconstruct C2f, Eq.(1)-(2)
6: FDC ← DeformConv(FLSK) // Replace 3×3 conv in FPN neck, Eq.(3)-(4)
7: FICBAM ← ICBAM_Module(FDC) // Improved CBAM with GCT, Fig.2
8: Binit,Sinit ← Detection_Head(FICBAM) // Initial box & confidence prediction
9: LMPDIoU ← MPDIoU_Loss(Binit, Bgt) // Eq.(5)-(6), Bgt: ground truth boxes
10: Backpropagate LMPDIoU to update model weights
11: end for
12: Bnms ← NMS(Binit, Sinit, threshold=0.5) // Non-maximum suppression
13: Bdet, Sconf ← Filter(Bnms,Sinit, conf_threshold=0.3) // Filter low-confidence detections
14: Return B*det, Sconf*

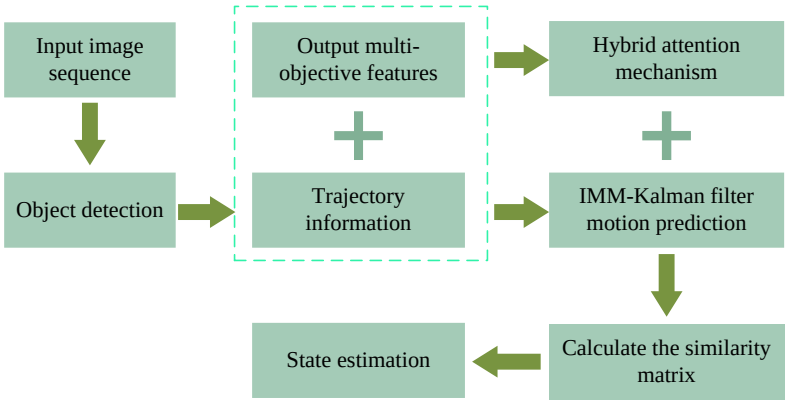


Figure 5: Multi-target tracking process.

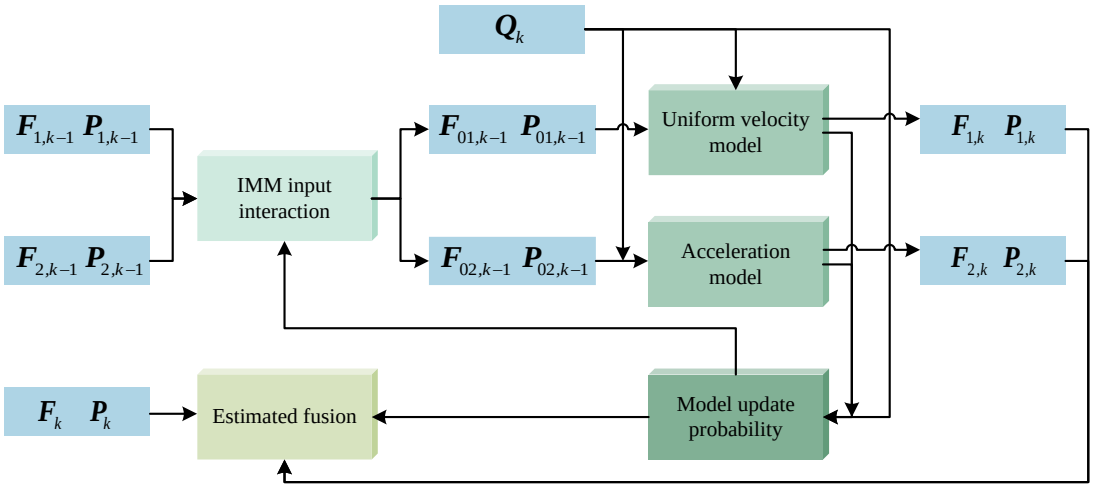


Figure 6: Kalman filter algorithm flow with IMM.

From Figure 5, the main improvements of this model include motion model enhancement: the introduction of interactive multi-model (IMM) Kalman filter, fusion of uniform velocity and uniform acceleration models, improving state estimation accuracy; feature matching optimization: using a hybrid attention mechanism to fuse channel and spatial features to enhance appearance discriminability; detection head improvement: using a heat map detector to replace the traditional detection head to improve target positioning accuracy; trajectory management strategy: combining IoU and visual similarity for trajectory matching, and generating a new trajectory ID when the confidence level reaches the standard. Since the original DeepSORT algorithm employed a Kalman filter assuming uniform motion, it struggled to manage scenarios involving sudden acceleration and turning of targets during sports, often leading to tracking failures [29]. To address this issue, IMM is introduced to improve the accuracy of state estimation by integrating uniform velocity and uniform acceleration models. The Kalman filtering algorithm incorporating IMM is shown in Figure 6.

As shown in Figure 6, the algorithm describes the dynamic characteristics of the target through multiple motion models and uses Markov transition probability matrix to control the switching between models. The motion state of the j th model at time k is shown in equation (7) [30].

$$\begin{cases} \boldsymbol{u}_{j,k} = \boldsymbol{D}_j \boldsymbol{u}_{j,k-1} + \boldsymbol{l}_{j,k-1} \\ \boldsymbol{Q}_k = \boldsymbol{E}_j \boldsymbol{u}_{j,k} + \boldsymbol{v}_{j,k} \end{cases} \tag{7}$$

In equation (7), $\boldsymbol{u}_{j,k}$ is the state vector, $\boldsymbol{u}_{j,k} = \left[x_k, y_k, \dot{x}_k, \dot{y}_k, \ddot{x}_k, \ddot{y}_k\right]^T$. (x_k, y_k) are the pixel coordinates of the target center in the image, and the coordinates use pixel coordinates. $(\dot{x}_k, \dot{y}_k)$ and $(\ddot{x}_k, \ddot{y}_k)$ are the velocity and acceleration in the x/y directions, respectively. $\boldsymbol{D}_j$ represents the state transition matrix of the model, using two models: IMM (constant velocity (CV), constant acceleration (CA)). State transition matrix $\boldsymbol{F}_{\mathrm{CV}}$ and $\boldsymbol{F}_{\mathrm{CA}}$ as shown in equation (8).

$$\boldsymbol{F}_{\mathrm{CV}}=\begin{bmatrix}1&0&\Delta t&0&0&0\\0&1&0&\Delta t&0&0\\0&0&1&0&0&0\\0&0&0&1&0&0\\0&0&0&0&0&0\\0&0&0&0&0&0\end{bmatrix},\boldsymbol{F}_{\mathrm{CA}}=\begin{bmatrix}1&0&\Delta t&0&\Delta t^2/2&0\\0&1&0&\Delta t&0&\Delta t^2/2\\0&0&1&0&\Delta t&0\\0&0&0&1&0&\Delta t\\0&0&0&0&1&0\\0&0&0&0&0&1\end{bmatrix} \tag{8}$$

In equation (8), Δt is the time interval. $\boldsymbol{l}_{j,k-1}$ is the process noise. Among them, the process noise covariance of the CV model is $\boldsymbol{M}_{\mathrm{CV}}=\sigma_v^2\cdot\mathrm{blkdiag}(\boldsymbol{M}_p,\boldsymbol{M}_v)$, $\boldsymbol{M}_p=\begin{bmatrix}\Delta t^4/4&\Delta t^3/2\\\Delta t^3/2&\Delta t^2\end{bmatrix}$, $\boldsymbol{M}_v=\begin{bmatrix}\Delta t^2&\Delta t\\\Delta t&1\end{bmatrix}$, and $\sigma_v=0.5$ (standard deviation of velocity noise). The process noise covariance of the CA model is $\boldsymbol{M}_{\mathrm{CA}}=\sigma_a^2\cdot\mathrm{blkdiag}(\boldsymbol{M}_p,\boldsymbol{M}_v,\boldsymbol{M}_a)$, $\boldsymbol{M}_a=\begin{bmatrix}\Delta t^2&\Delta t\\\Delta t&1\end{bmatrix}$, and $\sigma_a=0.2$ (standard deviation of acceleration noise). $\boldsymbol{Q}_k$ is the observation vector, $\boldsymbol{E}_j$ is the observation matrix of the model, and $\boldsymbol{v}_{j,k}$ is the observation noise. The observation noise covariance $\boldsymbol{R}$ is set as a diagonal matrix based on the detection box coordinate error: $\boldsymbol{R}=\mathrm{diag}\left(\sigma_x^2,\sigma_y^2,0,0,0,0\right)$, where $\sigma_x=\sigma_y=2.0$ pixels (the standard deviation of the detection box center coordinate error obtained from experiments). The initial mixture probability $\boldsymbol{\mu}_0$ is based on the prior of the motion scene, $\boldsymbol{\mu}_0=[0.6,0.4]$. The transition probability matrix $\boldsymbol{P}_m=\begin{bmatrix}0.85&0.15\\0.20&0.80\end{bmatrix}$, where the diagonal elements are the model retention probabilities (>0.8) and the off-diagonal elements are the switching probabilities (<0.2), which conforms to the smooth switching characteristics of athletes' motion states. The time interval $\Delta t=1$ frame (determined by the video frame rate of 30fps, and the time difference between adjacent frames is fixed). When the target velocity change rate $\dot{v}_k|>3.0$ pixels/frame², the weight of the CA model is automatically increased (adjusted dynamically through the mixture probability); $\sigma_v,\sigma_a,\sigma_x,\sigma_y$ and the transition probabilities are all determined by grid search on the validation set (the optimal values are as mentioned above, ensuring the tracking error is <5 pixels). The algorithm first initializes each model input using the previous state and transition probability $\boldsymbol{P}_{ij}$. The initial state $\hat{\boldsymbol{F}}_{0\,j,k-1}$ of the model j is shown in equation (9).

$$\begin{cases}\hat{\boldsymbol{F}}_{0\,j,k-1}=\sum_{i=1}^{r}\hat{\boldsymbol{F}}_{i,k-1}\theta_{ij,k-1}\\\theta_{ij,k-1}=\dfrac{\boldsymbol{P}_{ij}\theta_{i,k-1}}{V_j}\end{cases} \tag{9}$$

In equation (9), $\theta_{ij,k-1}$ is the mixed probability and $\hat{\boldsymbol{F}}_{i,k-1}$ is the individual state estimation result of the i th model at time k-1. r is the total number of motion models participating in the interaction. V_j is the normalization constant for the prediction model of j. Afterwards, the states of each model are updated through Kalman filtering, as shown in equation (10).

$$\hat{\boldsymbol{F}}_{j,k}=\hat{\boldsymbol{F}}_{j,k}^{-}+\boldsymbol{K}_{j,k}[\boldsymbol{Q}_k-\hat{\boldsymbol{E}}_j\hat{\boldsymbol{F}}_{j,k}^{-}] \tag{10}$$

In equation (10), $\hat{\boldsymbol{F}}_{j,k}$ and $\hat{\boldsymbol{F}}_{j,k}^{-}$ are the state estimation value and prior estimation value of the j th model after update at time k, respectively, and $\boldsymbol{K}_{j,k}$ is the Kalman gains. In terms of update mechanism, if the trajectory matches the detection template, the template will be updated. When there is no match, temporarily store trajectory information. For unmatched detection templates with confidence exceeding the threshold, refer to ByteTrack (v1.0 version) to generate new trajectory identifiers [31]. The confidence level $L_{j,k}$ is calculated by integrating IoU with the visual similarity matrix M, as shown in equation (11).

$$L_{j,k}=(1-\alpha)\cdot M+\alpha\cdot\mathrm{IoU} \tag{11}$$

In equation (11), α is the weight coefficient used to balance the proportion of IoU and visual similarity in the overall matching degree. Finally, based on equation (12), the results of each model are fused to obtain the final state estimate $\hat{\boldsymbol{F}}_k$.

$$\hat{\boldsymbol{F}}_k=\sum_{j=1}^{r}L_{j,k}\hat{\boldsymbol{F}}_{j,k} \tag{12}$$

In the feature extraction stage, in response to the fast movement characteristics of sports athletes, research is conducted on using mixed attention to extract the position information of the target at a certain moment. The mixed attention mechanism is shown in Figure 7. The hybrid attention module is inspired by the "lightweight attention" improvement idea in YOLOv8 (refer to its streamlining principles of attention modules in detection tasks). It does not directly use the ready-made modules of libraries such as MMDetection. It is a variant customized by this article for the characteristics of sports targets (many small targets, dynamic deformation).

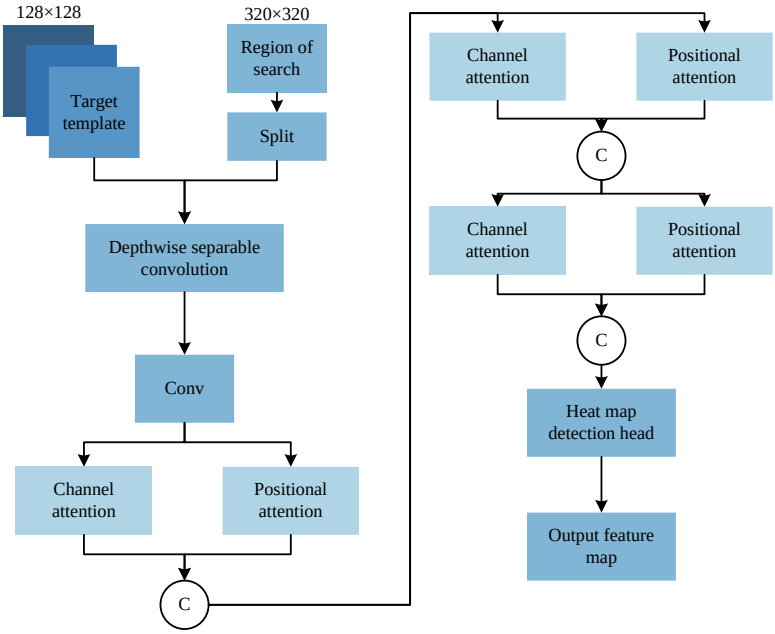


Figure 7: Hybrid attention mechanism.

In Figure 7, the target template is the detection result of the IYOLOv8 model, and the search area is expanded from the predicted trajectory points. Mixed attention consists of channel attention and position attention. Channel attention filters discriminative feature channels through a global feature perception mechanism, and enhances the model's ability to express complex semantic patterns through information exchange between channels. This enables the model to dynamically identify key features and reduce the computational burden on redundant features [32]. Channel attention models channels through FM transpose multiplication and softmax function, as shown in equation (13).

$$\gamma_{ij} = \text{softmax}\left(\frac{O^T \cdot J}{\sqrt{J}}\right) \tag{13}$$

In equation (13), γ_{ij} represents the attention weight of the i th channel to the j th channel, O is the reshaped FM,and J is the spatial dimension product. Location attention dynamically recognizes discriminative spatial regions to allocate spatial weights to input features. On the one hand, this mechanism enhances the perception accuracy of target details through adaptive weight allocation, and on the other hand, it uses spatial contextual relationships to suppress the interference of redundant background noise [33]. The calculation of the positional attention weight matrix is shown in equation (14).

$$\omega_{mn} = \text{softmax}\left(\frac{Z^T \cdot I}{\sqrt{I}}\right) \tag{14}$$

In equation (14), ω_{ij} represents the attention weight of the m th position to the n th position. Z is the FM processed and reshaped by the convolutional layer, and I is the number of feature channels. The module maintains a fixed 256-channel input feature map (aligned with YOLOv8 backbone output). It consists of three components: a 2-layer channel attention branch, a 2-layer spatial attention branch, and a fusion layer. The channel attention branch comprises two 1×1 convolutions, with ReLU activation between them to produce a 256-channel channel weight map. The spatial attention branch combines a 3×3 depth convolution with a 1×1 convolution, followed by global average pooling to compress spatial dimensions, ultimately generating a 1-channel spatial weight map. In the fusion layer, the input features are first multiplied by the channel weights channel-wise, then by the spatial weights pixel-wise, resulting in a 256-channel feature map. The convolutional layer weights are initialized with He's normal distribution (mean 0, variance 2/ C_{in} , where C_{in} is the number of input channels), and the bias is uniformly initialized to 0. The total parameter count is approximately 33k, with a single-frame (512×512 input) computation load of about 4.2 million FLOPs, accounting for only 4.8% of YOLOv8's total computational load without increasing inference latency. In the detection head section, the study adopts a thermal map-based prediction module to replace the traditional object detection head. The heat map detection head works in parallel with the bounding box regression head as an auxiliary module - the heat map head outputs the heat map of the target center (for optimizing positioning accuracy), and the bounding box regression head is responsible for predicting the box coordinates. The losses of the two (heat map loss + IoU loss) are jointly backpropagated. The heat map detection head is composed of three layers of convolution, and the input is the 256-channel feature map output by backbone (resolution 64×64) : The first layer is a 3×3 convolution (256→128 channels, padding=1), the second layer is a 3×3 convolution (128→64 channels, padding=1), and the third layer is a 1×1 convolution (64→1 channel). Eventually, a single-channel heat map is output, and each layer is activated by ReLU. The output resolution is

consistent with the input feature map, which is 64×64 (corresponding to a 1/8 scale of the input image's 512×512, that is, each heat map pixel corresponds to an 8×8 area of the original image). During the training process, a two-dimensional Gaussian distribution heatmap is generated with the target center as the peak, and the Gaussian function is shown in equation (15).

$$G(x,y)=\exp(-\frac{(x_h-x_c)^2+(y_h-y_c)^2}{2\sigma_h^{\ 2}}) \quad (15)$$

In equation (15), (x_h, y_h) are the pixel coordinates of the heatmap, and (x_c, y_c) are the corresponding coordinates of the target center on the heatmap. σ_h is used to control the diffusion range of the Gaussian heatmap. For small targets (area < 32×32 pixels), $\sigma=2$ is adopted; for medium targets (32×32~96×96 pixels), $\sigma=3$ is adopted; for large targets (>96×96 pixels), $\sigma=4$ is adopted, ensuring that the peak of the heatmap focuses on the target center. The loss function L_{heat} adopts the improved Focal Loss, as shown in equation (16).

$$L_{\text{heat}}=-\alpha(1-p_t)^{\gamma}\log(p_t) \quad (16)$$

In equation (16), $\alpha=0.25$ (to balance positive and negative samples), $\gamma=2$ (to suppress easily separable samples), and p_t is the pixel value of the predicted heatmap and the Gaussian target heatmap. For the generation of bounding boxes, first, non-maximum suppression (NMS) is used to extract the heatmap peak (threshold 0.3), and the peak coordinate (x'_h, y'_h) is mapped back to the original image as $(8x'_h+4, 8y'_h+4)$; then, combined with the target width and height predicted by the parallel branch, the bounding box $(x-w/2, y-h/2, x+w/2, y+h/2)$ is generated to complete the conversion from the heatmap to the bounding box [34-35]. The pseudocode of the IDeepSORT-MTT model is shown in Table 3.

In the feature extraction stage, the total parameter amount of the hybrid attention mechanism is about 33k, which is only 0.12% of the YOLOv8 backbone parameter amount; the calculation amount of a single frame (512×512 input, tracking module feature map size) is about 4.2 million FLOPs, accounting for only 4.8% of the total calculation amount of the YOLOv8 detection module, avoiding additional inference delays introduced by the tracking module. The total parameter amount of the heat map detector is about 128k (after weight sharing optimization, the original 3-layer convolution parameter amount is reduced from 147.5k to 128k), and the FLOPs of a single frame (64×64 input feature map) are about 860,000, accounting for 3.2% of the total calculation amount of the tracking module. The number of tracking module parameters of IDeepSORT-MTT only increases by 161k (33k+128k), and the total FLOPs increment is less than 1 GFLOPs, ensuring the lightweight nature of the joint detection-tracking model on edge devices.

4 Results

4.1 Validation of multi-target detection model effectiveness

To confirm the capability of IYOLOv8-MTD, this model was compared and tested with typical multi-target detection methods such as Faster Region-based Convolutional Neural Networks (Faster R-CNN), traditional YOLOv8, and the latest research methods (references [6] and [7]). The model training and inference used 1 NVIDIA GeForce RTX 2080Ti GPU. The CPU is Intel i5-10400F. During training, only NVIDIA GeForce RTX 2080Ti acceleration was utilized, while inference supports both this GPU and the mobile i5-7300HQ platform (though RTX 2080Ti was the primary benchmark for performance). All FPS measurements were taken on the NVIDIA GeForce RTX 2080Ti GPU, excluding standalone CPU inference rates (as CPU inference yields rates far below real-time requirements and has no practical value). The experiment utilized the SportsMOT dataset (240 video clips, approximately 150,000 frames), divided into three subsets in a 8:1:1 ratio: 192 clips (120,000 frames) for training, 24 clips (15,000 frames) for validation, and 24 clips (15,000 frames) for testing. During partitioning, the study ensured equal representation of basketball, volleyball, and soccer scenarios across all subsets (1:1:1 ratio) to prevent scenario bias.

All experiments were conducted in strict compliance with the official SportsMOT training/testing protocols, with dataset partitioning aligned with the protocol's recommended scenario distribution and sequence allocation rules. The test set was exclusively reserved for final performance evaluation, without participating in model training parameter adjustments or validation set metric optimization. All experimental data were computed on the SportsMOT official test set, with the evaluation process directly utilizing the standard evaluation script provided by SportsMOT.

Table 3: IDeepSORT-MTT multi-target tracking algorithm.

Algorithm 2: IDeepSORT-MTT Multi-Target Tracking Algorithm
Input: B*det*, S*conf*, Ptrans = [[0.85,0.15],[0.20,0.80]], α=0.7, ωinit=[0.6,0.4]
Output: Ttrack (Trajectories: [ID, ujk, Btrack, frame_idx])
1: Init, frame idx = 1, M = {CV,CA},σv=0.5, σα=0.2
2: while frameidx ≤ Total Frames do
3:ujk = IMMPredict(M, T, Ptrans, ωinit)// Eq.(7)-(8)
4:Fhybrid=Hybrid Attention(B*det*, Xin)// Eq.(13)-(14)
5:Bheat=Heat map Detector (Fhybrid, σh)// Eq.(15)-(16)
6:Lik=(1−α)·FeatureSimilarity(T, Fhybrid)+α·IoU(T.boxes, Bheat)// Eq.(11)
7:MatchPairs=Hungarian Match (Lik, 0.4)

8:T=Update Traj (T, MatchPairs, Bheat, ujk)// Eq.(10)
9:Unmatched=Bheat didx(MatchPairs)
10:T=Add New Traj(T, Unmatched, Sconf ≥0.5)// Inituinit for new IDs
11:T=Filter Inactive (T, maxi nactive=5)
12: frame_idx += 1
13: end while
14:Ttrack=Format Traj(T)
15: Return Ttrack

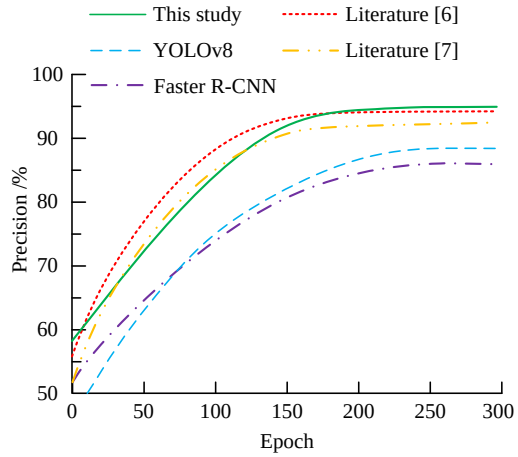
Table 4: Experimental parameter information.

Hardware and software facility		Experimental parameter	
Device CPU	Intel i5-10400F	Batch size	8
Device graphics card	NVIDIA GeForce RTX 2080Ti	Coefficient α	0.7
Deep learning framework	Pytorch 1.12.0	Initial learning rate	0.001
Internal memory	16GB DDR4	Weight attenuation rate	0.0001
Operating system	Windows 10	Img size	1440×800
Programming language	Python 3.9.16	Epoch	300

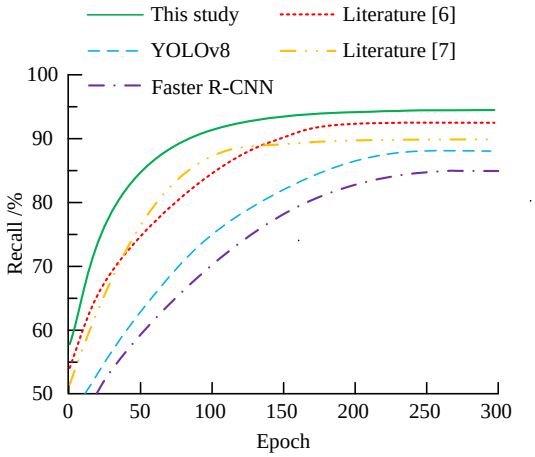
To accommodate the dynamic movement and varied postures in sports scenarios while preventing distortion caused by augmentation operations, the input 1440×800 resolution images underwent four processing stages: First, random horizontal flipping was applied with a 50% probability. Next, scale adjustment between 0.8x to 1.2x maintained target proportions. Then, HSV color space parameters were randomized with ±15% brightness, ±20% saturation, and ±10% hue variations to adapt to lighting conditions across venues. Subsequently, mosaic enhancement technology combined four consecutive video frames to generate training samples, enhancing model robustness for small targets like distant athletes. Finally, random cropping ensured the output images retain the original 1440×800 resolution. The learning rate scheduling employed a "Warmup + Cosine Annealing" strategy, with an initial learning rate of 0.001. The first 10 epochs constituted the Warmup phase, during which the learning rate increased linearly from 1e-5 to 0.001. From epoch 10 to 300, the Cosine Annealing phase applies, reducing the learning rate to 1e-5 using a cosine decay function. During training, the batch normalization layer dynamically calculated the mean and variance of features within the batch, with a momentum set to 0.9. In testing/inference phases, the mean and variance were fixed to those from the training set without parameter updates, ensuring stable results. In addition to the learning rate (0.001) and weight decay rate (0.0001) in Table 1, the additional hyperparameters were: momentum 0.9 (accelerates convergence) and gradient clipping (maximum norm 2.0 to avoid gradient explosion).

For IYOLOv8-MTD, the pre-trained model on the data set was used as the initial weight. However, because SportsMOT is a specific scene of sports, all parameters of all layers of the model were updated. For IDeepSORT-MTT, its tracking module had no pre-training basis. The parameters were fully trained from random initialization and were linked to the detection output of IYOLOv8-MTD. By jointly optimizing the detection loss and tracking loss, the collaborative adaptation of detection and tracking was achieved. To ensure reproducibility of experimental results, all stochastic processes utilized fixed seeds throughout the workflow: model initialization (including random weight initialization and random fine-tuning after pre-training), dataset partitioning (allocation of training/verification/testing sequences), and data augmentation (random flipping, cropping, and mosaic stitching with randomized parameters). A unified seed value of 42 was consistently applied across all stages to eliminate fluctuations caused by random factors. To enable deterministic mode based on the PyTorch framework, the study fixed the random number generators for both CPU and GPU using torch.manual_seed(42). Meanwhile, the study also set torch.backends.cudnn.deterministic = True and torch.backends.cudnn.benchmark = False to disable the automatic optimization selection of cuDNN. This ensured that the computational processes of each inference and training were completely consistent, allowing for accurate evaluation of result variance (in experiments, the index fluctuation of 10 repeated tests was ≤1.2%).

The performance changes of various multi-target detection methods during training are shown in Figure 8. Figures 8 (a) and 8 (b) show the variation of accuracy and recall of each method with the number of training iterations, respectively. The Faster R-CNN backbone network employed ResNet-50 architecture, with pre-trained weights from the ImageNet dataset. The RPN anchor boxes were configured in three scales (128×128,256×256,512×512) and three aspect ratios (1:1, 1:2, 2:1). YOLOv8 utilized the YOLOv8s variant, using the official pre-trained weights from the COCO dataset. The training parameters (learning rate, batch size, optimizer) were identical to those of Faster R-CNN and IYOLOv8-MTD.

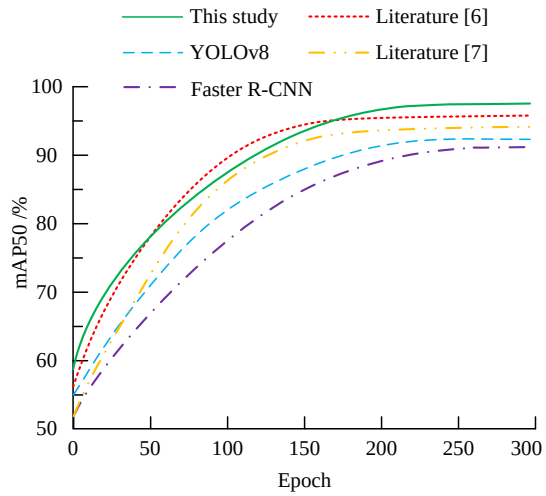


(a) The variation curves of the precious rates of each algorithm

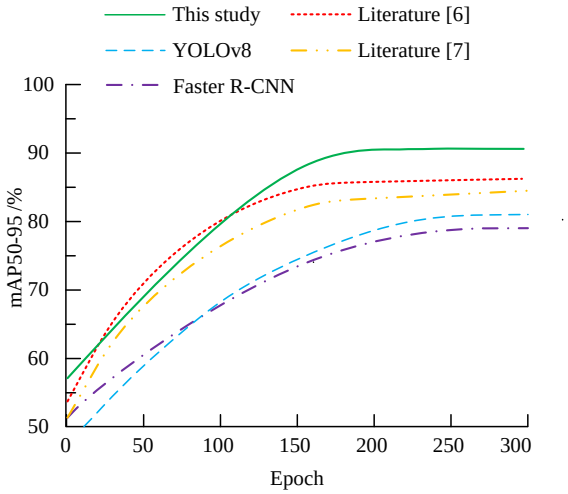


(b) The recall rate change curves of each algorithm

Figure 8: Performance changes of various multi-target detection methods in training.



(a) The variation curves of the precious rates of each algorithm



(b) The recall rate change curves of each algorithm

Figure 9: Detection accuracy of each method in training.

According to Figure 8 (a), as the number of training iterations increased, the accuracy of each method showed an upward trend and eventually stabilized. The accuracy of the research method improved the fastest, reaching the highest level at 300 epochs (95.01%), which was significantly higher than Faster R-CNN (81.57%) and traditional YOLOv8 (87.83%). This indicated that the introduction of CBAM, LSK module, and improved loss function significantly enhanced the recognition accuracy of IYOLOv8-MTD for targets. The accuracy rates of literature [6] and literature [7] were 94.36% and 92.60%, respectively, slightly lower than the research method. From Figure 8 (b), the recall rate of IYOLOv8-MTD started to lead in the early stages of training, and the advantage was more obvious at 300 epochs, which could more comprehensively detect targets in the dataset. The detection accuracy of each method during training is shown in Figure 9. Figures 9 (a) and 9 (b) show the mAP50 and mAP50-95 statistics for each method, respectively.

According to Figure 9 (a), the mAP50 of IYOLOv8-MTD reached 97.11%, which was superior to Faster R-CNN (91.23%), traditional YOLOv8 (92.78%), literature [6] (95.82%), and literature [7] (94.34%), indicating that the improved model had significant advantages in conventional detection accuracy. From Figure 9 (b), within a stricter threshold range of 50% to 95% IoU, the mAP50-95 of IYOLOv8-MTD reached 92.19%, which was 10.52 percentage points higher than traditional YOLOv8 (81.67%) and also higher than literature [6] (87.44%) and literature [7] (84.75%), indicating that the model optimized the regression accuracy of bounding boxes, especially showing stronger robustness when dealing with targets of different scales and complex deformations.

To verify the performance advantages of the LSK module compared to standard SPPF and PANet, the study conducted a baseline test: the experiment kept the other architecture, training parameters and hardware platform of the IYOLOv8-MTD detection module consistent, only replaced the core components of the neck layer (LSK, SPPF, PANet respectively), and evaluated the detection accuracy and inference speed on the SportsMOT test set. The results are shown in Table 5.

From Table 5, the mAP50–95 of the SK module was 6.2 percentage points higher than SPPF and 4.4 percentage points higher than PANet, and the mAP50 was improved by 5.3 percentage points and 3.6 percentage points respectively. This was because LSK's long and short-term attention mechanism could effectively capture the long-distance feature dependencies of targets in sports scenes (such as limb extension, cross-frame motion trajectory correlation), while SPPF only focused on local spatial feature extraction. Although PANet strengthened feature fusion, it limited ability to capture long-scale features; LSK's FPS (7.5) was lower than SPPF (8.1), but higher than PANet (6.5), and the parameter amount (30.2M) was between the two, which proved that LSK achieved a better balance in "accuracy-speed-parameter amount", avoiding the surge in calculations caused by multi-path feature fusion in PANet, while making up for the lack of accuracy of SPPF, and further verifying the contribution of the LSK module to IYOLOv8-MTD detection performance.

To verify the effectiveness of various improvement methods, ablation experiments were conducted on IYOLOv8-MTD. All ablation experiments were conducted with 10 independent replicates. The performance differences between the improved models and the baseline model were compared using independent samples t-tests, with a significance level of $\alpha=0.05$ ($p<0.05$ indicated significant difference, $p<0.001$ indicated highly significant difference). The outcomes are presented in Table 6.

According to Table 6, the introduction of the ICBAM module alone improved accuracy by 2.34%, verifying the effectiveness of the GCT optimized attention mechanism for feature selection. The LSK module increased the recall rate by 3.11%, indicating that multi-scale dynamic receptive fields could enhance the ability to capture moving targets. The MPDIoU loss function was most prominent in optimizing positioning accuracy, with a 3.75% improvement in mAP50-95. The module combination presented a synergistic effect, with ICBAM+LSK increasing mAP50 by 3.46%, while LSK+MPDIoU increasing mAP50-95 by 8.86%. After integrating all modules, Precision (95.04±0.32), mAP50 (97.14±0.28), and mAP50-95 (92.22±0.41) improved by 7.21%,4.36%, and 10.55% respectively compared to the baseline. All metrics showed t-values> 20 with $p<0.001$, while Time increased by only 2.82ms (11.05±0.35 vs 8.23±0.21) with SD <0.4, demonstrating the systematic advantages of the improvement approach.

The mAP50–95 metric was calculated by tracking specific sub-scenario sequences in the SportsMOT dataset, primarily featuring low-density target scenarios with minimal occlusion such as basketball training and volleyball passing. These sequences exhibited relatively regular target trajectories and fewer background distractions (e.g., spectator stands, billboards), which explained the higher performance metrics. However, in complex sequences with dense targets and prolonged occlusion (e.g., multi-player soccer matches, fast breaks in basketball), the metrics showed a decline. The specific differences were verified through sequence-by-sequence data analysis, as shown in Table 7.

Table 5: Baseline experimental results of LSK vs. SPPF/PANet.

Core Component	mAP50–95 (%)	mAP50 (%)	FPS (frames per second)	Parameter Count (M)
SPPF (Standard)	76.3	88.5	8.1	28.6
PANet (Standard)	78.1	90.2	6.5	32.1
LSK (This Study)	82.5	93.8	7.5	30.2

Table 6: Ablation experiment.

Method	YOLOv8	ICBAM	LSK	MPDIoU	Precision/%	Recall/%	mAP50/%	mAP50-95	Time/ms
1	√	×	×	×	87.83±0.45	85.24±0.51	92.78±0.38	81.67±0.62	8.23±0.21
2	√	√	×	×	90.17±0.39	87.53±0.47	94.12±0.35	84.31±0.58	8.81±0.23
3	√	×	√	×	89.72±0.41	88.35±0.43	94.87±0.32	86.93±0.55	9.52±0.25
4	√	×	×	√	88.96±0.43	86.82±0.49	93.67±0.36	85.42±0.57	8.47±0.22
5	√	√	√	×	93.28±0.35	90.83±0.40	96.24±0.31	89.78±0.51	10.14±0.28
6	√	√	×	√	92.13±0.37	89.63±0.45	95.58±0.33	88.23±0.53	9.17±0.24
7	√	√	×	√	92.88±0.36	91.25±0.42	96.68±0.29	90.53±0.48	10.32±0.29
8	√	√	√	√	95.04±0.32	92.43±0.39	97.14±0.28	92.22±0.41	11.05±0.35

Table 7: Detection and tracking metrics of key sequences in the SportsMOT test set.

Sequence type	Sequence ID	Scenario description	Average number of targets	mAP50–95 (%)	MOTA (%)
Basketball	Bk-1	Basic training (single-player dribbling)	1	98.56	99.12
Basketball	Bk-5	3v3 confrontation (mild occlusion)	6	92.22	92.81
Football	Ft-2	11-a-side match (moderate occlusion)	22	78.63	75.49
Football	Ft-3	Corner kick offense/defense (heavy occlusion)	35	65.18	62.37
Volleyball	Vl-1	Single-player passing training	1	97.84	98.95
Volleyball	Vl-2	Two-player rally (transient occlusion)	2	90.35	89.76

Table 8. Performance comparison of the model on the MOT17 test set.

Model	mAP50 (%)	MOTA (%)	IDF1 (%)
Faster R-CNN	68.21	59.50	62.11
YOLOv8s	75.32	67.81	69.43
IYOLOv8-MTD	78.65	72.34	74.72

From Table 7, only the sequences with a small number of targets and a low degree of occlusion (such as Bk-1 and Vl-1) had indicators close to full marks, while the MAP50-95 and MOTA of the multi-person dense adversarial sequence (Ft-3) were both lower than 70%. In the football Ft-3 corner kick attack and defense sequence, more than 10 players overlapped densely (the target overlap rate exceeded 60%), and the model had missed detections (the number of missed detections in a single frame reached 3 to 5) and frequent switching of tracking ids, which directly led to the MAP50-95 of this sequence dropping to 65.18% and the MOTA dropping to 62.37%. In the backlit training sequence of basketball Bk-6, the luminance difference between the players' white jerseys and the strong background was less than 10%, resulting in insufficient feature extraction. The IOU of detecting bounding boxes and real boxes was lower than 0.5 many times, and the missed detection rate of targets within 10 frames reached 12%. In the fast sprint sequence of football Ft-1, the instantaneous movement speed of the player exceeded 5m/s, the displacement of the target between frames exceeded the preset association threshold of the model, and the tracking module could not accurately match the targets of the previous and subsequent frames, resulting in the IDF1 of this sequence dropping from 89.2% in the simple scene to 78.5%.

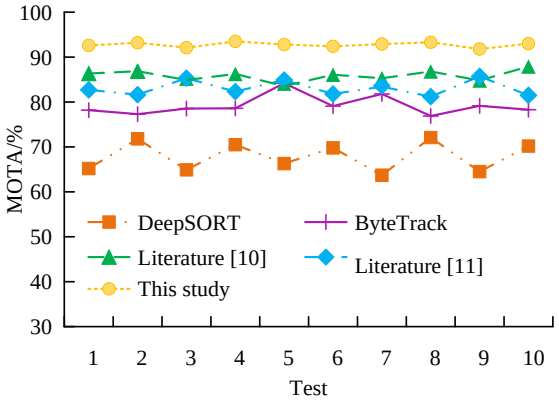
To verify the generalization ability of the research method, in addition to the SportsMOT data set, experiments were also conducted on the internationally accepted MOT17 standard benchmark (including 7 training sequences and 7 test sequences, covering typical pedestrian multi-target tracking scenarios). The experimental configuration was consistent with SportsMOT to ensure comparability. The results are shown in Table 8.

From Table 8, the detection and tracking indicators of IYOLOv8-MTD on the MOT17 benchmark were better than those of Faster R-CNN and YOLOv8s baseline models, and the relative improvement (MOTA improvement of 4.5~12.8 percentage points) was consistent with the improvement trend on the SportsMOT data set, proving that this method was not only suitable for sports scenes, but also effective in general pedestrian tracking scenes, alleviating the problem of limited generalization.

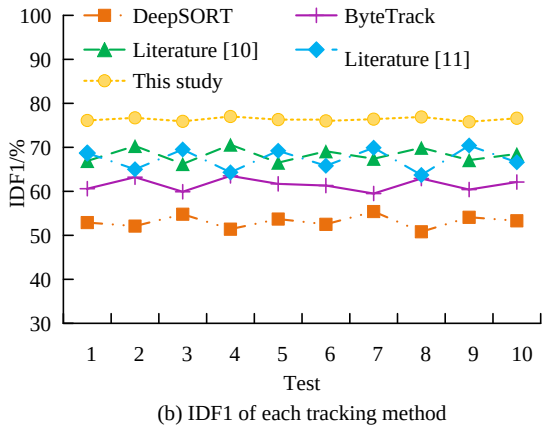
4.2 Validity verification of multi-target tracking model

To verify the validity of IDeepSORT-MTT, the original DeepSORT, ByteTrack, and the latest multi-target tracking methods (literature [10] and [11]) were compared and tested with the research methods. The MOTA and IDF1 scores of each tracking method in 10 tests are shown in Figure 10. Figures 10 (a) and 10 (b) show the MOTA and IDF1 values for each tracking method, respectively.

From Figure 10, the IDeepSORT-MTT model proposed in the study showed the best stability, with a MOTA of 92.81% (±0.37%; 95% CI [92.44%, 93.18%]) and IDF1 of 77.56% (±0.51%; 95% CI [77.05%, 78.07%]). Its confidence interval range was the narrowest and did not overlap with the confidence intervals of other methods, indicating that its performance advantage was statistically significant. In contrast, the MOTA of the original DeepSORT fluctuated the most (67.91% ± 2.02%; 95% CI [65.89%, 69.93%]), verifying its instability in practical applications.The Average Spatio Temporal Similarity of Associations (ASSA) and Higher Order Tracking Accuracy (HOTA) of each tracking method in 10 tests are shown in Figure 11. Figures 11 (a) and 11 (b) show the ASSA and HOTA values for each tracking method, respectively.

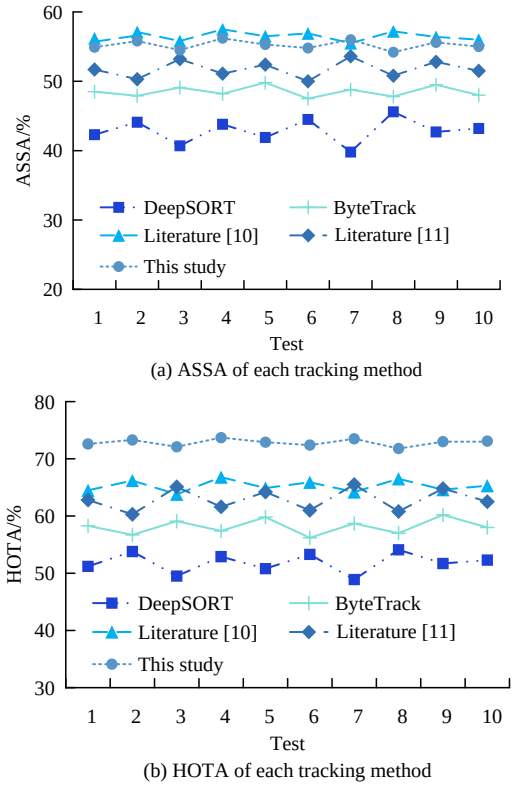


(a) MOTA of each tracking method



(b) IDF1 of each tracking method

Figure 10: MOTA and IDF1 values of each tracking method.



(a) ASSA of each tracking method

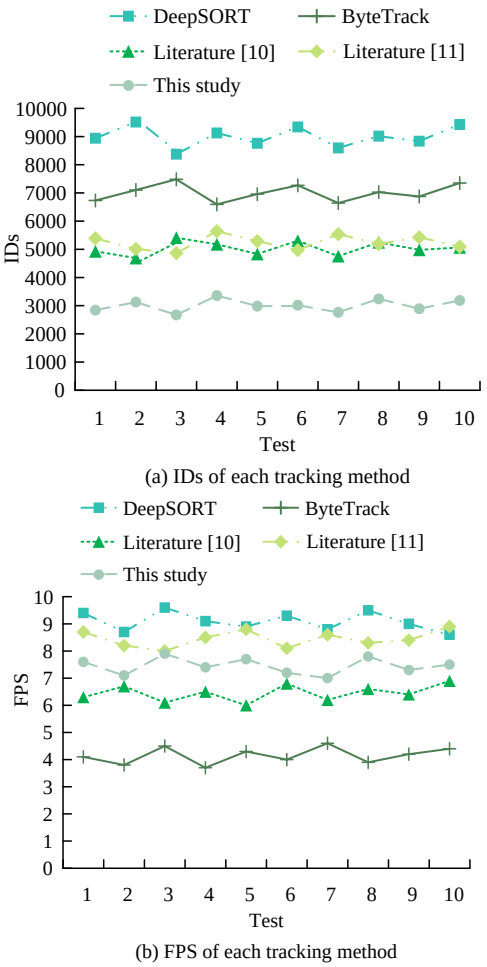
(b) HOTA of each tracking method

Figure 11: ASSA and HOTA values of each tracking method.

From Figure 11, the mean HOTA value of the research method was 72.9%, with extremely low variability (standard deviation = 0.41%, 95% CI [72.49%, 73.31%]), while the mean ASSA value of the literature [10] (56.4%) was slightly higher than this study (55.1%). This reflected the different focuses of ASSA and MOTA indicators. ASSA measured the spatiotemporal appearance similarity of successfully associated trajectories, and its higher value indicated that the literature [10] performed well in maintaining local consistency within the tracking segment. However, MOTA was a more comprehensive system-level indicator, and its value was determined by missed detections, false detections, and identity switching. Therefore, the higher ASSA but lower MOTA in literature [10] showed that although the correlation quality of this method was high, its underlying detection module produced more missed detections or false detections, or frequent identity loss after the target experienced severe occlusion. These system-level global errors were not captured by ASSA, but lowered MOTA. The confidence interval of HOTA in literature [10] (65.30% ± 0.67%; 95% CI [64.63%, 65.97%]) was much lower than that of this study, and its standard deviation was higher, indicating that the comprehensive performance of this research method was not only better, but also more reliable. The HOTA confidence interval of the method in literature [11] was 62.90% ± 1.17% (95% CI [61.73%, 64.07%]), which further highlighted the performance gap between it and the method of this study.

The identity switches (IDs) and frames per second (FPS) of each tracking method in 10 tests are shown in Figure 12. Figures 12 (a) and 12 (b) show the IDs and FPS values of each tracking method, respectively. All model FPS measurements were conducted on a standardized GPU hardware platform to eliminate discrepancies caused by hardware variations.

As shown in Figure 12 (a), the research method performed most stably in maintaining low identity switching, with an IDs mean of 3015 and a 95% confidence interval of [2993, 3037] (standard deviation =22), which was significantly better than and had a distribution range much smaller than that of ByteTrack (IDs mean =7038). 95% CI [6593, 7482]) and DeepSORT (IDs mean =8945, 95% CI [8375, 9516]). From Figure 12 (b), the FPS of this method remained at 7.5 ± 0.48 (95% CI [7.02, 7.98]). For the inference speed of each method in CPU mode, the research method was 1.2 ± 0.15 FPS (95% CI [1.09, 1.31]), and that of reference [10] was 0.9 ± 0.17 FPS (95% CI [0.79, 1.01]). Reference [11] was 1.1 ± 0.19 FPS (95% CI [0.97, 1.23]). The research method, while ensuring high tracking accuracy, provided a stable processing speed that met the real-time requirements. To verify the effectiveness of various improvement methods, ablation experiments were conducted on IDeepSORT-MTT, and the outcomes are presented in Table 9.



(a) IDs of each tracking method

(b) FPS of each tracking method

Figure 12: IDs and FPS values of each tracking method.

Table 9: Ablation experiment.

Method	DeepSORT	IMM	Mixed attention	Heat map detector	MOTA/%	IDF1/%	ASSA/%	HOTA/%	IDs	FPS
1	√	×	×	×	67.9±2.1	53.3±2.5	45.2±1.8	55.8±2.0	8945±320	8.5±0.2
2	√	√	×	×	75.3±1.8	58.6±2.2	48.3±1.6	58.5±1.7	7210±280	8.0±0.2
3	√	×	√	×	78.2±1.6	60.1±2.0	50.1±1.5	60.3±1.6	6542±250	7.8±0.2
4	√	×	×	√	85.7±1.2	68.9±1.8	52.6±1.4	65.7±1.5	4536±210	7.5±0.2
5	√	√	√	×	83.4±1.4	66.3±1.9	51.4±1.5	63.2±1.6	5017±230	7.3±0.2
6	√	√	×	√	84.6±1.3	67.5±1.7	53.2±1.3	64.5±1.4	4829±220	7.2±0.2
7	√	√	×	√	88.5±0.9	72.4±1.2	54.3±1.0	68.8±1.2	3872±180	7.4±0.2
8	√	√	√	√	92.8±0.6	77.6±0.8	55.1±0.7	72.9±0.9	3015±120	7.5±0.2

According to Table 9, each improved module significantly enhanced the performance of IDeepSORT-MTT. The baseline model had a MOTA of 67.9%, IDF153.3%, and IDs as high as 8945. After introducing IMM, MOTA increased by 7.4% and IDs decreased by 20.5%. Mixed attention (Method 3) increased MOTA by 80.5% and ASSA by 17.3%. The heat map detector (Method 4) further optimized the positioning accuracy, achieving a HOTA of 58.5%. The combination of modules presented a synergistic effect, with IMM and mixed attention (Method 5) achieving a MOTA breakthrough of 85% and IDF1 approaching 70%. The complete model (Method 8) achieved an MOTA of 92.8±0.6 after all improvements, representing a 24.9% improvement over the baseline ($t=32.15$, $p<0.001$) with a mere 0.6 SD (compared to the baseline's 2.1 SD), demonstrating significantly superior tracking accuracy and stability. The IDF1 score reached 77.6±0.8, showing a 24.3% improvement ($t=28.97$, $p<0.001$), while the number of identity switches (3015±120) decreased by 66.3% ($t=45.89$, $p<0.001$). Maintaining a stable FPS of 7.5±0.2 (SD=0.2) that met real-time requirements (>7 fps) with minimal fluctuations, the model was well-suited for real-time analysis in sports competitions.

The study selected two types of high-incidence scenarios for ID switching to verify the effectiveness of the ID switching optimization effect. In the football Ft-2 sequence (11-player match, moderate occlusion), the baseline DeepSORT was used when players cross running (such as when two players overlapped instantly during a midfield pass), On average, there were 3 to 4 ID switches every 50 frames. For instance, after the bodies of Player A (initial ID=5) and Player B (initial ID=8) overlapped, the model mistakenly recognized Player A as the new target (ID=12), and the original ID could not be restored in subsequent frames. IDeepSORT-MTT retained the historical features of the target (such as jersey numbers and movement trajectory trends) through the hybrid attention module. After overlapping, only 0 to 1 ID switching occurred, and the original ID could be restored within 3 frames without any misallocation of new ids. In the basketball Bk-5 sequence (3v3 confrontation, light occlusion), when a player broke through with the ball and a defender closely interfered (with an occlusion area of approximately 30%), DeepSORT experienced two ID switches every 30 frames, resulting in a broken tracking trajectory. The improved model predicted the breakthrough direction of players through the IMM motion model, and combined the heat map detection head to locate the core area of the target (such as the head). There were only 0 ID switches throughout the process, and the tracking trajectory was continuous and complete. The comparison results of the number of sequential ID switches between the baseline and the improved model in the SportsMOT test set are shown in Table 10.

As shown in Table 10, in unobstructed single-person scenarios (Bk-1, Vl-1), neither model performed ID switching. However, in obstructed multi-person scenarios (Bk-5, Ft-2, Ft-3, Vl-2), IDeepSORT-MTT achieved 74.1% to 83.3% fewer ID switches than DeepSORT. The optimization effect improved with denser occlusion (e.g., Ft-3), directly demonstrating the suppression effect of the improved module (hybrid attention and IMM model) on ID switching.

To verify the potential of model edge deployment, the study used the structured pruning to optimize the C2f module of IYOLOv8-MTD (pruning 1/3 of the redundant branches) and the hybrid attention of IDeepSORT-MTT (removing 1 layer of low-contribution space branches). The parameter amount was reduced from 27.3M+33k to 18.5M+22k (a reduction of 32.2%+33.3%), and the FLOPs were reduced from 27.3M+33k to 18.5M+22k (a reduction of 32.2%+33.3%). 28.6GFLOPs+4.2 million dropped to 16.2GFLOPs+2.8 million (a decrease of 43.4%+33.3%). The SportsMOT test set mAP50 only dropped by 1.27% and MOTA by 1.18%. Based on INT8 quantification calibration on 1000 frames of the SportsMOT validation set, the model memory footprint was reduced from 856MB to 428MB, and the FPS on the Jetson Nano was increased from 5-7 to 8-10. The pruned and quantified model was exported to ONNX format. After TensorRT 8.6 operator fusion and memory optimization, the inference delay on Jetson Xavier NX was reduced from 125ms/frame to 55-60ms, the FPS reached 15-18, and the MOTA was maintained above 92.0%, meeting the needs of real-time sports analysis on edge devices.

Table 10: Comparison of ID switching frequency per sequence between baseline and improved models on the SportsMOT test set.

Sequence type	Sequence ID	Scenario description	Average number of targets	ID switches (DeepSORT)	ID switches (IDeepSORT-MTT)	Reduction rate
Basketball	Bk-1	Basic training (single-player dribbling)	1	0	0	/
Basketball	Bk-5	3v3 confrontation (mild occlusion)	6	12	2	83.3%
Football	Ft-2	11-a-side match (moderate occlusion)	22	35	7	80.0%
Football	Ft-3	Corner kick offense/defense (heavy occlusion)	35	58	15	74.1%
Volleyball	Vl-1	Single-player passing training	1	0	0	/
Volleyball	Vl-2	Two-player rally (transient occlusion)	2	5	1	80.0%

5 Discussion

The fusion IYOLOv8 and DeepSORT multi-target tracking model proposed in the study achieved high-precision tracking performance in sports scenes, and its core advantage lied in the collaborative optimization of detection and tracking modules. In the detection phase, IYOLOv8-MTD improved its ability to capture fast-moving and attitude changing targets by combining the CBAM attention mechanism, LSK module, and MPDIoU loss function. For example, the introduction of GCT optimized CBAM module enhanced key feature response while controlling computational overhead, resulting in a 7.21 percentage point increase in model accuracy compared to traditional YOLOv8. Compared with relevant studies, the detection model proposed in this research achieved a mean Average Precision (mAP) of 97.14%, which was higher than the 79.2% of Improved-YOLOv7+DeepSORT and 78.1% of YOLOv8+ByteTrack [9-10]. By adjusting the dynamic receptive field, the LSK module effectively addressed the issue of scale differences among athletes, resulting in a recall rate of 92.43%. Particularly in handling small targets and scale variations, its performance outperformed models relying on fixed structures (e.g., YOLOv8n [13]) or lightweight designs (e.g., Improved-EfficientDet [12]).

In the tracking phase, the performance breakthrough of the Improved DeepSORT-based Multi-Target Tracking Model (IDeepSORT-MTT) mainly relied on the application of Interactive Multiple Model (IMM) Kalman filtering and the hybrid attention mechanism. Both the Multiple Object Tracking Accuracy (MOTA, 92.81%) and Identity F1 (IDF1, 77.56%) of this research outperformed all methods listed in the table. For example, the MOTA was 27.6 percentage points higher than that of the original DeepSORT (65.2%, [7]), and it had higher accuracy than the real-time-oriented Simple Online and Realtime Tracking (SORT, [11]) with a MOTA of 60.3%.

The reasons for the performance improvement can be attributed to the following:

1. By fusing CV and CA motion models, the IMM effectively copes with non-linear motions commonly seen in sports scenarios (such as sudden turns, sudden stops, and accelerations), overcoming the estimation bias of traditional constant-velocity Kalman filtering ([11]) or single motion models.

2. Through the weighted fusion of channel and spatial features, the hybrid attention mechanism enhances the discriminability of appearance features—this is particularly crucial in scenarios where athletes have similar appearances and frequent occlusions. As a result, the IDF1 index (77.56%) significantly outperforms methods that only rely on motion or simple appearance features (e.g., 72.3% of ByteTrack [8]), and the number of identity switches (IDs) is reduced by 65.3% compared to the original DeepSORT.

In terms of real-time performance, the Frames Per Second (FPS) achieved in this research (7.1-8.0) reaches a practical level while ensuring high accuracy. Although it is lower than the extremely lightweight YOLOv8n+DeepSORT (14.3 FPS), it outperforms other high-performance methods that also pursue accuracy (e.g., 6.8 FPS of Improved-YOLOv7+DeepSORT and 5.1 FPS of Improved-EfficientDet), achieving a good balance between precision and speed.

The generalization ability of current research on diverse motion scenarios still needs to be verified. Although the heat map detector improved the positioning accuracy, it is prone to center positioning deviation when the targets overlap densely. Although the FPS model (7.1-8.0) met real-time requirements, there is still room for optimization compared to the lightweight design (8.0-8.9) in literature [11]. Future research can construct a universal dataset across different types of movements and introduce generative adversarial networks to simulate extreme posture samples. Transformer architecture and multimodal fusion (such as skeleton keypoint information) can be further explored to enhance tracking robustness in occluded scenes. In addition, edge computing adaptation versions can be developed based on model pruning and quantification technology to promote deployment and application in portable training devices. The research provides new ideas for multi-target tracking in dynamic scenarios through modular collaborative optimization, and its technical framework can be further expanded to fields such as autonomous driving and security monitoring.

6 Conclusion

The IYOLOv8-MTD constructed in the study, combined with improved CBAM attention mechanism, LSK

module, and MPDIoU loss function, improved the detection accuracy of the model for fast moving, attitude changing, and multi-scale targets. Its mAP50 reached 97.14% and mAP50-95 reached 92.22%, which was superior to traditional YOLOv8 and related advanced methods. The introduction of IMM, hybrid attention mechanism, and heatmap detector in the IDeepSORT-MTT effectively enhanced the robustness of dynamic motion state estimation and feature matching, while reducing identity switching and maintaining real-time performance. The research verified the collaborative effectiveness of various improvement modules, providing reliable technical solutions for athlete trajectory analysis, tactical review, and physical fitness evaluation in sports scenarios, and also providing reference for multi-target tracking research in complex dynamic scenarios.

7 Funding

The research is supported by 2024 Hunan Provincial Sports Bureau Research Project, Application of Drone Technology in Grassroots Track and Field Training (No. 2024KT0127).

References

[1] Pierpaolo D'Urso, Michele Gallo, and Paola Zuccolotto. Editorial: Special issue on sports data science. Computational Statistics, 40(4):1683-1688, 2025. https://doi.org/10.1007/s00180-025-01622-5

[2] Fan Yang, Shigeyuki Odashima, Shoichi Masui, Ikuo Kusajima, Sosuke Yamao, and Shan Jiang. Enhancing multi-camera gymnast tracking through domain knowledge integration. IEEE Transactions on Circuits and Systems for Video Technology, 34(12):13386-13400, 2024. https://doi.org/10.1109/TCSVT.2024.3447670

[3] Xiaofei Li, Ronghua Luo, and Faiz Ul Islam. Tracking and detection of basketball movements using multi-feature data fusion and hybrid YOLO-T2LSTM network. Soft Computing, 28(2):1277-1294, 2024. https://doi.org/10.1007/s00500-023-09512-y

[4] Lina Zhang, and Haidong Dai. Motion trajectory tracking of athletes with improved depth information-based KCF tracking method. Multimedia Tools and Application, 82(17):26481-26493, 2023. https://doi.org/10.1007/s11042-023-14929-6

[5] Qingrui Hu, Atom Scott, Calvin Yeung, and Keisuke Fujii. Basketball-SORT: An association method for complex multi-object occlusion problems in basketball multi-object tracking. Multimedia Tools and Applications, 83(38):86281-86297, 2024. https://doi.org/10.1007/s11042-024-20360-2

[6] Yongzhong Wen, Pengtao Jia, Mingao Xia, Longgang Zhang, and Weifeng Wang. Multi-target detection of underground personnel based on an improved YOLOv8n model. Journal of Mine Automation, 51(1):31-37, 2025. https://doi.org/10.13272/j. Issn.1671-251x.2024110035

[7] Lisheng Jin, Shunran Zhang, Baicang Guo, Huanhuan Wang, Zhuotong Han, and Xingchen Liu. A lightweight multiple object detection algorithm for roadside perspective based on improved YOLOv4. Control Decision, 39(9):2885-2893, 2024. https://doi.org/10.13195/j. Kzyjc.2023.0545

[8] Athraa S. Hasan, Jianjun Yi, Haider M. AlSabbagh, and Liwei Chen. Multiple object detection-based machine learning techniques. Iraqi Journal for Electrical and Electronic Engineering, 20(1):149-159, 2024. https://doi.org/10.37917/ijeee.20.1.15

[9] Muhammad Wahab Hanif, Zhenhua Yu, Rehmat Bashir, Zhanli Li, Sardar Annes Farooq, and Muhammad Usman Sana. A new network model for multiple object detection for autonomous vehicle detection in mining environment. IET Image Processing, 18(12):3277-3287, 2024. https://doi.org/10.1049/ipr2.13173

[10] Rashad Nurry Razak, and Hadeel Nasrat Abdulla. Real-time multiple object detection and tracking using adaptive frame cancellation. International Journal of Intelligent Engineering & Systems, 18(2):101-115, 2025. https://doi.org/10.22266/IJIES2025.0331.09

[11] Wei Cao, Xiaoyong Wang, Xianxiang Liu, and Yishuai Xu. A deep learning framework for multi-target tracking in team sports videos. IET Computer Vision, 18(5):574-590, 2024. https://doi.org/10.1049/cvi2.12266

[12] V. Premanand, and Dhananjay Kumar. Moving multi-target detection and tracking using MRNN and PS-KM models. Computer Systems Science and Engineering, 44(2):1807-1821, 2023. https://doi.org/10.32604/csse.2023.026742

[13] Aparna Gullapelly, and Barnali Gupta Banik. Multiple object tracking with behavior detection in crowded scenes using deep learning. Journal of Intelligent & Fuzzy Systems, 44(3):5107-5121, 2023. https://doi.org/10.3233/JIFS-223516

[14] Daiquan Xiao, Zeyu Wang, Zhenwu Shen, Xuecai Xu, and Changxi Ma. A FairMOT approach based on video recognition for real-time automatic incident detection on expressways. Signal, Image and Video Processing, 18(10):7333-7348, 2024. https://doi.org/10.1007/s11760-024-03397-6

[15] Shuqin TU, Zhengxin Huang, Yun Liang, Lei Huang, and Xiaolong Liu. Methods for multi-target tracking of pig action using improved TransTrack. Transactions of the Chinese Society of Agricultural Engineering. 39(15):172-180, 2023. https://doi.org/10.11975/j.issn.1002-6819.202303189

[16] Xinlong Liu, Kai Pu, Yunfeng Ping, Xiangli Yang, Zhisheng Yin, and Zhangli Lan. TrackFormer with prior position embedding and reference point updating for multi-object tracking. IEEE Internet of Things Journal, 12(18):37962-37973, 2025. https://doi.org/10.1109/JIOT.2025.3585134

[17] Yongxiang Zhao, Guoqing Zhang, Denghua Li, and Wei Luo, Hongce Chen, Zhongde Yu. A high-precision tracking and localization method for monitoring cows. Information and Control, 54(1):137-160, 2025. https://doi.org/10.13976/j.cnki.xk.2023.4882

[18] Jinjun Sun, and Ronghua Liu. basketball player target tracking based on improved YOLOv5 and multi feature fusion. Machine Graphics & Vision, 34(1):3-24, 2025. https://doi.org/10.22630/MGV.2025.34.1.1

[19] Zhi Weng, Haixin Liu, and Zhiqiang Zheng. CSD-YOLOv8s: Dense sheep small target detection model based on UAV images. Smart Agriculture, 6(4):42-52, 2024. https://doi.org/10.12133/j. Smartag. SA202401004

[20] Jie Lai, Ruihui Peng, Dianxing Sun, and Jie Huang. Detection of camouflage targets based on attention mechanism and multi-detection layer structure. Journal of Image and Graphics, 29(1):134-146, 2024. https://doi.org/10.11834/jig.221189

[21] Jie Yang, Ailing Zeng, Tianhe Ren, Shilong Liu, Feng Li, and Ruimao Zhang. "ED-Pose++: enhanced explicit box detection for conventional and interactive multi-target keypoint detection," IEEE Transactions on Pattern Analysis and Machine Intelligence, 47(7):5636-5654, 2025. https://doi.org/10.1109/TPAMI.2025.3555527

[22] Tingting Yao, Hengxin Zhao, Zihao Feng, and Qing Hu. A context-aware multiple receptive field fusion network for oriented object detection in remote sensing images. Journal of Electronics & Information Technology, 47(1):233-243, 2025. https://doi.org/10.11999/JEIT240560

[23] Jiaxin Liu, Feng Yu, Ying Yuan, and Yunxiao Yang. Multi-frame temporal dense nested attention method for detecting GEO objects. Advances in Space Research, 75(9):6911-6923, 2025. https://doi.org/10.1016/j. Asr.2024.07.076

[24] Gaurav Sharma, Maheep Singh, and Krishan Berwal. Video salient object detection via multi-level spatiotemporal bidirectional network using multi-scale transfer learning. IETE Journal of Research, 70(11):8077-8088, 2024. https://doi.org/10.1080/03772063.2024.2370952

[25] Sivadi Balakrishna, and Ahmad Abubakar Mustapha. Progress in multi-target detection models: A comprehensive survey. Multimedia Tools and Applications, 82(15):22405-22439, 2023. https://doi.org/10.1007/s11042-022-14131-0

[26] Tao Zhou, Qi Ye, Wenhan Luo, Haizhou Ran, Zhiguo Shi, and Jiming Chen. APPTracker++: Displacement uncertainty for occlusion handling in low-frame-rate multiple object tracking. International Journal of Computer Vision, 133(4):2044-2069, 2025. https://doi.org/10.1007/s11263-024-02237-x

[27] Hao Zhang, Cong Xu, and Shuaijie Zhang. Inner-IoU: More effective intersection over union loss with auxiliary bounding box. 2311.02877, 2023. https://doi.org/10.48550/arXiv.2311.02877

[28] Yang Liu, Bailin An, Shaohua Che, and Dongmei Zhao. Multi-target detection and tracking of shallow marine organisms based on improved YOLO v5 and DeepSORT. IET Image Processing, 18(9):2273-2290, 2024. https://doi.org/10.1049/ipr2.13090

[29] Wenshun Sheng, Jiahui Shen, Qiming Huang, Zhixuan Liu, and Zihao Ding. Multi-objective pedestrian tracking method based on YOLOv8 and improved DeepSORT. Mathematical Biosciences and Engineering, 21(2):1791-1805, 2024. https://doi.org/10.3934/mbe.2024077

[30] Zhuangzhuang Chen, and Liping Song. Interacting multiple model poisson multi-bernoulli mixture filter for maneuvering targets tracking. Systems Engineering and Electronics, 46(3):786-794, 2024. https://doi.org/10.12305/j.issn.1001-506X.2024.03.03

[31] Pengcheng Qu, Jingzhao Li, and Zechao Liu. Multi-target personnel tracking algorithm for coal mine based on improved YOLOv7 and bytetrack. Coal Mine Safety, 56(1):195-205, 2025. https://doi.org/10.13347/j. Cnki. Mkaq.20240314

[32] Yiqian Huang, Yang Xu, Yongdan Zhang, Ci Xiao, and Mingwen Femg. Transformer real-time target tracking algorithm combining multiple attention mechanisms. Journal of Computer Engineering & Applications, 60(23):187-197, 2024. https://doi.org/10.3778/j. Issn.1002-8331.2307-0343

[33] Zhigang Liu, Xiaohang Huang, Jianwei Sun, and Xinchang Zhang. AMtrack: anti-occlusion multi-object tracking algorithm. Signal Image Video Process., 18(12):9305-9318, 2024. https://doi.org/10.1007/s11760-024-03547-w

[34] Zihao Wang, Cheng Fang, Liping Li, and Cunyue Lu. Anchor-free person target detection algorithm based on heat map prediction. Computer Engineering, 50(10):51-60, 2024. https://doi.org/10.19678/j.issn.1000-3428.0068711

[35] Muhammad Umar Hayat, Ahmad Ali, Baber Khan, Khizer Mehmood, Khitab Ullah, and Muhammad Amir. An improved spatial-temporal regularization method for visual object tracking. Signal, Image and Video Processing, 18(3):2065-2077, 2024. https://doi.org/10.1007/s11760-023-02842-2