Novel Human Object Detection Method Based on YOLO Deep Learning Optimization Strategy

Ting Zhang [1], Shuqi Niu [2], Lu Chen [2], Chenhui Dou [2], Tao Liu [2,*], and Degan Zhang [2,*]

[1] School of Sports Economics and Management, Tianjin University of Sport, Tianjin 301617, China
[2] Tianjin Key Lab of Intelligent Computing and Novel software Technology, Tianjin University of Technology, Tianjin 300384, China
Email: 2285246377@qq.com (T.Z.); 2840082717@qq.com (S.N.); 1287725598@qq.com (L.C.); 3209134406@qq.com (C.D.); 44128592@qq.com (T.L.); 2310674826@126.com (D.Z.)
*Corresponding author

Abstract—**Along with the development of volleyball match video analysis, this paper introduces a novel human target detection method based on an optimized You Only Look Once (YOLO) deep learning strategy. Firstly, shot segmentation (identifying scene boundaries) is performed on the volleyball videos to extract key frames. Then, using semantic annotation techniques (classifying segments to filter non-game content), the videos are described as sequences of shots composed of long shots, medium shots, close-ups, replays, and off-court shots. Secondly, to address the issue of slow speed in target detection algorithms, the backbone network of YOLOv8s is optimized by implementing lightweighting through the GhostNet network and enhancing semantic information with the Convolutional Block Attention Module (CBAM) module to improve model accuracy. Experimental results on the PascalVisual Object Classes (PASCAL VOC), Common Objects in Context (COCO), and the Volleyball datasets, as well as real volleyball match videos, demonstrate that the proposed algorithm achieves a 1.5% higher mAP@50 with 64.2% fewer computational load Giga Floating- point Operations Per Second (GFLOPs) compared to the baseline YOLOv8s, achieving an optimal balance between accuracy and efficiency, making it suitable for real-time tactical analysis and automated player performance statistics in coaching and broadcasting.**

Keywords—**semantic annotation, object detection, deep learning, lightweight network, volleyball video**

I. Introduction

With the rapid development of computer vision technology, sports video analysis has become an active research area. Particularly in volleyball, recognizing and analyzing athletes and their movements can significantly enhance the efficiency of training and the scientific nature of match strategies. However, due to the fast movements and complex posture changes of volleyball players, automatic detection and analysis of events in videos face many challenges.

Early volleyball video event detection mainly relied on traditional machine learning methods. These methods utilized prior information from images, such as low-rank properties and temporal correlations, and employed matrix or tensor decomposition or Bayesian representations, combined with optimization algorithms, to obtain the final fusion results. However, these traditional methods were limited by high computational complexity or sub-optimal feature representation, which seriously affected their practical performance and applicability. Later, deep learning methods were applied to volleyball video event detection [1–6]. Compared with traditional machine learning methods, deep learning methods demonstrated superior performance, with significant improvements in accuracy and robustness.

One research direction in computer vision is object detection, whose core task is to identify and locate objects in images or videos. Human object detection is a more specific branch of this task, which is crucial for understanding people in images and serves as the foundation for more advanced applications, such as pose estimation and action recognition [7–12].

Deep learning-based object detectors, such as You Only Look Once (YOLOv8), demonstrate superior performance in general applications. However, their direct application to volleyball match analysis introduces several unique challenges. First, live broadcasting and real-time tactical analysis demand exceptionally high processing speed. Although YOLOv8s is relatively efficient, its standard version still involves a substantial number of parameters and significant computational overhead, which may limit deployment on resource-constrained devices or in scenarios requiring ultra-high frame rates. Second, the volleyball court environment presents complex visual conditions, including frequent player occlusions, transitions between long shots and close-ups, large variations in player scale, and cluttered backgrounds with crowd interference [13–16]. These factors necessitate a model with enhanced capabilities for multi-scale object detection and robust attention to salient features.

Manuscript received August 20, 2025; revised October 16, 2025; accepted October 27, 2025; published January 20, 2026.

doi: 10.12720/jait.17.1.153-170

Therefore, there is a pressing need to develop a specialized human detection method that achieves an optimal balance among accuracy, inference speed, and model compactness for volleyball video analysis.

In contrast to generic YOLO optimization approaches, this work proposes a dedicated volleyball analysis framework that integrates video semantic understanding with an optimized detection architecture. The key innovations include: 1) a lightweight GhostNet backbone to improve computational efficiency [17–19], 2) Convolutional Block Attention Module (CBAM) modules to enhance feature representation in complex scenes [20–22], and 3) Sturcture Intersection over Union (SIoU) loss to improve localization accuracy—integrated within a semantically aware processing pipeline [23–25]. This holistic approach is specifically tailored to address the distinctive challenges of sports video analysis, including real-time processing demands, frequent occlusions, and significant variations in object scale.

We identify all relevant event video clips from match videos and perform human object detection on the pre-processed video results to recognize the players' posture information for subsequent action analysis and statistics, such as analyzing the volleyball serving action. Therefore, there are high requirements for the accuracy and real-time performance of video processing and human object detection algorithms [26–28].

Videos are composed of continuous image frames, which contain additional temporal information compared to static images. In volleyball match videos, there are twelve athletes, which poses a challenge to human pose estimation algorithms, especially when dealing with multi-person scenarios. How to apply techniques originally designed for single individuals to multi-person video scenarios is the focus of this research [29–31]. This chapter specifically studies this issue and proposes video shot segmentation to obtain key video frames and human object detection on video frames. The innovation of this method of this paper lies in its optimization of a single-stage object detection network structure, simplification of the backbone network to increase processing speed, and introduction of a new loss function to accelerate the model training process and improve model performance [32–38]. The algorithm flow for player detection in volleyball videos is shown in Fig. 1 below.

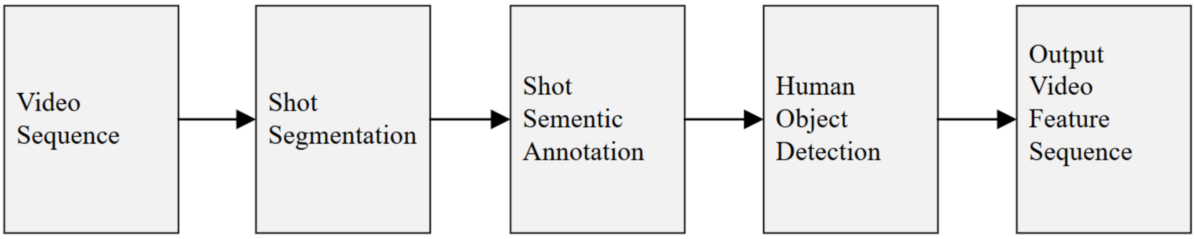


Fig. 1. Flowchart of player detection in volleyball videos.

Fig. 1 presents the overall architecture of the proposed volleyball video analysis framework. The pipeline comprises three main stages [39–45]: 1) video preprocessing, where the original footage undergoes shot segmentation (dividing the video into contiguous segments) and key frames are extracted to substantially reduce data redundancy; 2) semantic shot annotation (categorizing key frames into types such as long shots, medium shots, and replays), to filter out non-game content and prioritize computational resources for relevant game segments; and 3) optimized player detection using an enhanced YOLOv8s model, which is applied exclusively to selected game shots [46–50]. By integrating video semantics with object detection, this workflow forms the foundation of our approach, ensuring efficient and accurate analysis [51–55].

Moshayedi *et al.* [56] studied the integration of Autonomous Aerial Vehicles (AAVs) has significantly advanced image processing and remote sensing, particularly in precision agriculture. This study addresses the challenge of accurately quantifying corn production by developing an enhanced YOLO-v8-based deep learning model, incorporating dynamic and fixed labeling techniques, tested on 810 images and video data for real-time detection. The research utilized two primary datasets totaling 570 images. The evaluation process comprised four distinct tests: conducted on Dataset 1 with 200 images, assessed seven attention mechanisms (Spatial Extended Attention (SE), CBAM, Gobal Attention (GA), Local Key Attention (LKA), Channel Attention (CA), Soft attention (SA), and Time Attention (TA)) using deep learning metrics (Precision, Recall, mAP50, mAP50-95, F1-score) and statistical methods. This study advances computer vision in agriculture, offering a scalable, high-accuracy model for corn yield estimation, with broad applications in farming optimization, financial planning, and policy-making.

And they analyzed the components and design features of robots employed in corn fields. This analysis not only serves as a comparison tool for designers but also encourages the development of more diverse designs. The structure of robots in corn farming plays a crucial role in advancing agricultural practices by boosting efficiency, precision, adaptability, data collection capabilities, environmental sustainability, and safety standards [57]. They introduced a comprehensive approach to detecting and analyzing ammonia in agricultural settings. It elucidates the merits and demerits of conventional indoor and outdoor ammonia detection methods, juxtaposing them with the innovative technology of Electronic nose (E-nose) and seven widely employed ammonia detection methods in farmland are scrutinized and compared against traditional techniques. They did comprehensive comparative analysis encompassing all the aforementioned methodologies, elucidating the potential and limitations of E-nose in facilitating ammonia detection endeavors within agricultural contexts [58].

In existing literature, deep learning algorithms are considered powerful technologies that have demonstrated remarkable performance and effectiveness in various fields. For example, in image recognition, natural language processing, and medical diagnosis, deep learning algorithms, by mimicking the structure of the human brain's neural networks, are capable of processing and analyzing vast amounts of data to achieve highly complex tasks. Moreover, deep learning algorithms continue to advance, and with in-depth research, they show greater potential in solving even more complex problems. However, in this context, their application in volleyball is relatively limited and many drawbacks, such as fast mobility, more security, more reliability, and so on.

In order to solve the above issues (the research objectives or aim and motivation of this paper), the main contributions of this paper are as follows:

A decision-tree-based algorithm for volleyball video frame segmentation is proposed. We develop an efficient preprocessing pipeline for volleyball game video that integrates shot segmentation and semantic annotation to filter irrelevant frames and focus computational resources on key game segments.

The YOLOv8s architecture specifically for human detection in sports videos is optimized, and a lightweight structure (GhostNet) is introduced to reduce parameters and computational cost, and an attention mechanism (CBAM) is introduced to enhance feature representation under complex backgrounds.

The loss function has been improved to enable the model to fit the data better, thereby accelerating the convergence speed and reducing the training time.

The proposed method is comprehensively verified on public datasets (PASCAL VOC, COCO) and a specialized volleyball dataset as well as real game videos. Compared with the existing state-of-the-art detectors, the proposed method is superior in accuracy, speed, and model complexity.

II. Literature Review

For the analysis of volleyball match videos, it is necessary to first process the videos by extracting key frames. Only then can existing deep learning algorithms be used to identify the positions of players and subsequently assess their postures and actions.

A. Event Detection Methods

The task of video event detection is to extract segments with specific significance from long videos, with the aim of automatically identifying abnormal events using computer technology [7]. In sports video analysis, this technology can filter out exciting moments that may interest the audience from the entire match video. Initially, this work was entirely manual, with professionals using video editing software to mark and edit the desired segments based on their experience. However, given the need for rapid processing and the large volume of sports videos, manual operations are not only time-consuming and labor-intensive but also lack accuracy.

Event detection in volleyball videos can be divided into two distinct techniques: one based on manually set rules and the other using machine learning. The latter trains models on annotated video segments to enable automatic event detection in new videos [8]. This method is highly automated and widely applicable, but in practice, machine learning faces challenges in collecting comprehensive training samples. Insufficient training data can affect the model's performance and applicability. In contrast, manually set rules require defining detection criteria based on the characteristics of volleyball videos, which necessitates a deep understanding of the video content. Overly simple rules may reduce accuracy and applicability, while overly complex rules can affect the stability and generalizability of the detection [9].

Considering practical application scenarios, a three-layer semantic structure analysis method can be adopted. This method includes the extraction of low-level visual features, recognition of middle-level entities and scenes, and analysis of high-level entity relationships and attributes. First, low-level semantic features are extracted through video shot segmentation. Next, these features are used to semantically annotate the shots, resulting in semantically labeled shots. Finally, events are detected by combining machine learning and manual rules.

B. Object Detection Algorithms

In the task of object detection, traditional methods are often disturbed by factors such as changes in lighting, cluttered backgrounds, varying object sizes, and occlusions [10]. These factors hinder the algorithm's ability to accurately capture the core features of the target, thereby affecting the detection accuracy and stability of the model [11]. In contrast, deep learning techniques can accurately extract target features in complex environments, making the model more robust [12–16].

There are two major categories of deep learning applications in object detection: one-stage and two-stage methods.

The Region-Convolutional Neural Networks (R-CNN) model was proposed by Girshick *et al.* [16] in 2014 and was the first object detection model to use deep learning. The model identifies potential object regions using the selective search method, which involves dividing the image into multiple blocks of varying sizes, merging similar blocks, and finally filtering out potential object regions. Since neural networks can only process images of fixed size, normalization is required. Subsequently, Convolutional Neural Networks (CNNs) are used to extract features from these regions, and the extracted features are classified using Support Vector Machines (SVMs) to determine the object category and predict its location in the image. Although R-CNN significantly improved detection performance, it has several limitations. Its training process is complex and time-consuming due to the need to generate a large number of potential object regions, which slows down both training and detection speeds. Additionally, resizing regions can cause image distortion, affecting detection accuracy.

Unlike two-stage algorithms, single-stage methods directly predict the target's location, confidence score, and category on the feature map, outputting results in one step. This approach eliminates the need for intermediate region proposal generation, resulting in faster detection speeds and reduced hardware requirements, making it more suitable for practical applications.

Redmon and Farhadi [21] proposed the You Only Look Once (YOLO) object detection algorithm, which was the first single-stage detection model that merged region proposal and object recognition into one stage. YOLOv1's detection logic is based on the grid cell containing the center of the object being responsible for detecting that object. The model divides the image into a 7×7 grid and predicts two bounding boxes for each grid cell, resulting in a total of 98 candidate boxes. It uses Intersection over Union (IoU) to filter the predicted boxes against the ground-truth boxes, with higher IoU values indicating better localization.

YOLOv1 employs a lightweight feature extraction network to achieve fast regression and classification. Although this approach improves detection speed, it has several limitations. Each grid cell predicts only two bounding boxes and one class, which restricts the number of detectable objects and weakens overall detection performance. Additionally, assigning equal loss weights to all objects results in lower accuracy when detecting small objects [19–23].

In 2024, Varghese and Sambath [24] proposed the latest model in the YOLO series, YOLOv8. This network adopts state-of-the-art backbone and neck structure designs and is equipped with an anchor-free, decoupled Ultralytics head. It achieves an ideal balance between accuracy and inference speed, making it highly suitable for real-time object detection tasks in various application scenarios [25].

C. Model Light-Weight Methods

Miniaturized models, with their smaller storage footprint and lower computational load, are more readily deployable in practical applications and can more effectively create value.

Deep learning has unique advantages in the field of image processing. However, complex deep convolutional neural networks also make real-time processing tasks such as object detection and pose estimation more challenging on GPUs. As the number of network layers increases, although model accuracy is improved, more computational power is required. Therefore, it is necessary to compress and accelerate the models, reducing their depth and computational load while maintaining detection accuracy. This can speed up model training and prediction, enabling algorithms to be more stably and efficiently deployed on other platforms. This is crucial for the practical application of deep learning technologies.

Using more streamlined CNNs can enhance model performance. Through optimization, models can achieve excellent real-time performance across different platforms. In the development of deep learning, several techniques aimed at reducing computation have emerged, such as Xception, the MobileNet series (including MobileNetV1, MobileNetV2, MobileNetV3), the Shuffle Net series (including Shuffle-NetV1, Shuffle NetV2), and GhostNet [26–32]. While Xception simplifies computation, it has higher requirements for GPU memory. The MobileNet series uses a large number of 1×1 convolutions and depthwise separable convolutions to streamline calculations. The ShuffleNet series not only employs grouped convolutions but also enhances channel information to improve model performance. GhostNet, on the other hand, considers the correlation and redundancy between feature maps, generating redundant features through linear transformations and intrinsic features through identity mappings. By combining these two approaches, GhostNet reduces the number of parameters and computational load.

Network structure lightweighting aims to reduce the number of model parameters without consuming more resources. This goal is typically achieved through techniques such as pruning and quantization.

In deep learning, both fully connected layers and convolutional layers contain a large number of unnecessary parameters that contribute little to model performance [33]. Moreover, many neurons have activation values close to zero, leading to wasted computational resources. Dropout, a technique that randomly deactivates some neurons, achieves a pruning effect by reducing the number of active neurons. Additionally, transforming fully connected networks into sparse networks is a common method for model compression [34–36].

Pruning involves evaluating the weights of network connections to determine whether to maintain connections between neurons, and then trimming neurons and their connections accordingly. This simplifies the network structure, reduces computational load, and speeds up training. However, pruning can lead to a drop in accuracy, which necessitates retraining and fine-tuning of the pruned model to recover lost precision [37–43].

Sparse models require the reuse of network parameters, which reduces the number of parameters without increasing computation, thereby shrinking the model size. For fully connected networks, random weight sharing methods can be used to compress the model while maintaining its performance, and connections with non-shared weights are associated [44–48].

III. Materials and Methods

In this section, we first use shot segmentation methods to process the video into groups of individual frames. Then, we extract key frames from the sequences to represent each shot group. Next, we perform semantic classification and labeling on the key frames. Finally, we detect all players in the images from key frames with specific semantic information. The detected candidate bounding boxes are then enlarged, cropped, and saved by position for subsequent input into pose estimation.

A. Shot Segmentation

Videos contain a vast amount of data. This paper chooses to remove the audio from the video, treating the video as a sequence composed of continuous frames. Processing event detection tasks frame by frame involves a significant amount of computation. To reduce the computational burden, frames with similar content are grouped together, and one or more key frames are extracted from each group as representatives. The video is then decomposed into a sequence of content-independent shots, which not only reduces the computational load but also ensures the integrity of the video content. To extract these frames, the video needs to be segmented into a series of meaningful and manageable video segments, namely shots. A shot is a collection of a series of interrelated video frames and constitutes the basic unit for event detection in volleyball videos.

A common approach to shot segmentation is to use boundary detection algorithms to determine the boundary frames of each shot in the video, and then segment the video into independent shots based on these boundary frames. This process typically involves calculating the feature differences between frames and setting a specific threshold to judge whether the changes between frames are significant. Once the threshold is exceeded, it can be considered that the current frame marks the beginning of a new scene, thereby achieving video scene segmentation. Shot segmentation is mainly divided into two types: one is pixel comparison, which is highly effective for videos with minor and gradual changes in the scene; the other is histogram analysis, which is suitable for detecting rapidly changing shots. Since volleyball match videos often involve frequent shot changes and relatively simple backgrounds, a pixel comparison-based segmentation algorithm is adopted. The frame difference calculation as Eq. (1):

$$D(k, k+1) = \frac{1}{HW}\sum_{x=1}^{H}\sum_{y=1}^{W}\left|I_k(x,y) - I_{k+1}(x,y)\right| \quad (1)$$

where, $D(k, k+1)$ represents the average absolute frame difference per pixel, which serves as the core metric for our shot segmentation algorithm. The factor $\frac{1}{HW}$ acts as a normalization term, ensuring that the calculated difference is the average per-pixel value rather than a total sum. The width and height of the video frame are denoted by W and H, respectively. $I_k(x,y)$ represents the brightness of the current frame at point (x, y), while $I_{k+1}(x,y)$ represents the brightness at the same position in the next frame. When the difference $D(k, k+1)$ exceeds a predefined threshold, it indicates that the frames belong to different scenes in the video.

We employs a dual-comparison method to determine the threshold for detecting scene changes, which can simultaneously identify both gradual and abrupt transitions. Initially, a higher threshold value is used to quickly detect abrupt changes in the video. Subsequently, a lower threshold value is applied to locate the starting frame of a gradual change, and the cumulative frame difference is calculated until the end frame of the gradual change is identified. When the cumulative value of the frame differences remains below the predefined threshold for an extended period, a tolerance value is set to allow a certain number of consecutive frames with minor differences before determining that no change has occurred.

After shot segmentation, a series of independent shot groups are obtained. However, the specific content contained in these segments is still unknown; they are merely units separated physically. Subsequently, it is necessary to analyze the content of these shot groups, that is, to examine each video frame within each shot group frame by frame. However, similar frames within the same group would be redundant. Therefore, it is necessary to extract one or more frames from each shot group as key frames to represent the entire shot group, thereby reducing the number of frames.

Technological advancements have brought about various key frame extraction techniques. Among them, shot-boundary-based extraction algorithms are suitable for shot groups segmented from complete videos, where the feature changes between adjacent frames are minimal. Visual-content-based extraction algorithms can comprehensively represent the content of a shot but are prone to extracting redundant key frames, making video processing time-consuming. Motion-analysis-based extraction algorithms introduce motion features to obtain more accurate key frames but are computationally complex. Clustering-based extraction algorithms can extract representative key frames but require predefined cluster numbers and centers, and they have longer running times.

Considering the characteristics of volleyball videos and the requirements for efficiency, as well as the Twin Comparison shot segmentation algorithm used in this paper, analysis of the shot groups after segmentation reveals that the differences between frames within each group are relatively small. After weighing the advantages and disadvantages of various key frame extraction algorithms, the shot-boundary-based key frame extraction algorithm was chosen. This algorithm can efficiently and effectively select key frames. Among the extracted key frames, the middle frame is typically chosen as the final key frame. The sequence key frame obtained according to the algorithm presented in this paper is shown in Fig. 2.

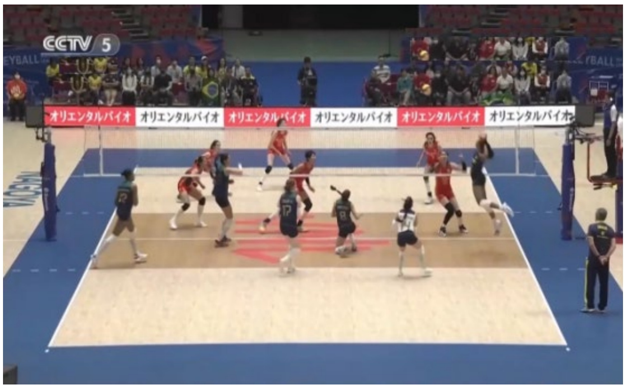


Fig. 2. The key frame of a sequence in a real match video.

B. Shot Semantic Annotation

For a computer, the segmented shot sequences are merely a series of images, but the meanings behind these images are not recognized by the computer. Therefore, it is particularly important to perform semantic labeling on key frames. The content of volleyball videos is relatively simple, so it is possible to analyze the feature values of key frames and use decision tree techniques to categorize the video into different classes and label semantic information.

Fig. 3 illustrates the rule-based decision tree developed for automatic semantic shot annotation. The classification follows a hierarchical logic grounded in computationally derived metrics. Initially, replay shots are identified through the detection of broadcast logos. Subsequent to this, non-replay shots are distinguished by their playfield ratio: a high PR value classifies a shot as a long shot, typically encompassing the majority of the court, whereas a medium PR indicates a medium shot. For shots with a low PR, the algorithm further evaluates the edge ratio. A low ER corresponds to a close-up shot, characterized by simple structural content such as a player's face, while a high ER suggests a complex external shot, often focusing on the audience or coaching staff. This structured, rule-based annotation is fundamental to automating video content interpretation and ensures that subsequent analytical processes, such as player detection, are concentrated on the most relevant game segments.

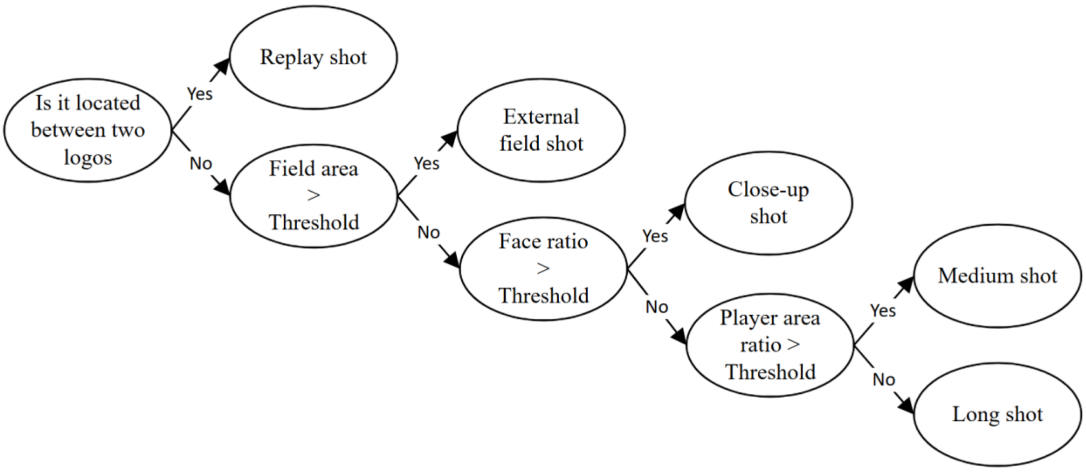


Fig. 3. Shot semantic labeling decision tree.

Replay shots play a guiding role in the detection and positioning of events in volleyball match videos. For viewers, replays not only enhance the viewing experience but also provide insights into unexpected incidents, with directors often replaying exciting details captured from different angles. When creating replay footage, to alert the audience, the competition's logo is usually added at the beginning and end of the footage, allowing viewers to naturally get into the mood and continue watching the match smoothly. The replay shots mentioned in this paper actually refer to a set of shots composed of two frames with logos and a series of normal or slow-motion frames, which, for convenience, are simply referred to as replay shots, and will not be elaborated on further.

Slow motion is often seen in replay footage, which can serve as a marker to identify replay shots. There are two main ways to create slow motion: one is to capture the action with a high-speed camera and then play it back at normal speed, allowing viewers to see slow motion; the other is to use a regular camera and simulate slow motion through technical means such as repeating frames or interpolating frames. Depending on the production method, different approaches can be used to detect replays. One is the template matching method. The template matching method first creates a template based on the replay frames and then uses the template to compare with the video frames that belong to the replay. However, this method has not been very accurate and is relatively time-consuming. The second is the logo detection method. This method detects replay shots by identifying logos inserted before and after the replay. To ensure a smooth viewing experience, replays are usually placed in the middle of the event logo, so replays can be indirectly determined by recognizing the logo. However, different events may have different logos, and one logo template cannot be used to detect replays in all videos. By studying a large number of volleyball match videos and related literature, this paper proposes using motion feature vectors to determine the pixel area of the logo, thus solving the non-universal problem. Experiments have proven that this method can be used in videos of major events such as the Olympics, World Championships, and FIVB Volleyball World Cup, with a high identification accuracy rate.

We identifies logo shots by comparing key frames of suspected logo shots with preset logo templates; shots that fall within a certain similarity threshold are considered logo shots. Fig. 4 shows the key frames of event logos obtained using the logo detection method with motion feature vectors, and replays are typically located between two logo key frames.

The long shots and medium shots in the text refer specifically to the court scenes captured by the camera. Long shots can capture the entire or most of the court scene; medium shots show one or several players in full body and posture, and although the court is visible, it occupies a much smaller proportion of the frame compared to long shots. Therefore, the two types of shots can be distinguished by using the Playfield Ratio (PR) metric.

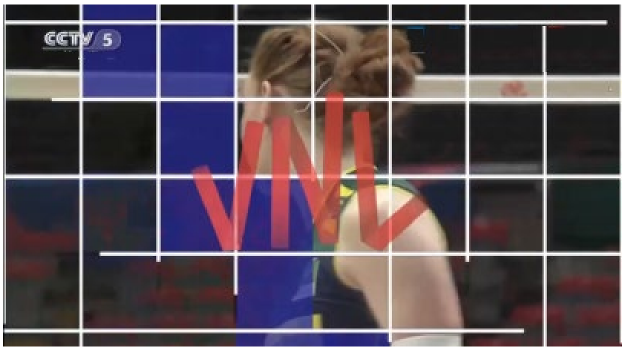


Fig. 4. Key frames of replay shots in real match videos.

To calculate the Playfield Ratio, it is necessary to identify the color of the court. However, there is no uniform standard for the color of volleyball courts; they are only required to be light-colored and different from the boundary line color, so each court has slightly different color values. Long shots include court scenes, but partial audience seats may appear and cause errors. To address this, this paper reduces the impact on the court ratio by normalizing the top one-third of the pixels of all frames to be detected before calculating the main color range of the court. The steps for algorithm implementation are as follows:

1) Before the match starts, capture the scene of the players entering the court, using it as a key frame to determine the court color. By calculating the pixel distribution, select the color corresponding to more than 50% of the pixels as the court color.

2) Normalize only the bottom two-thirds of the area of non-replay key frames, and then calculate the Playfield Ratio (PR). The calculation formula for PR as Eq. (2):

$$PR(j) = \frac{D_1(j)}{w \times h \times \frac{2}{3}} \tag{2}$$

where, $D_1(j)$ represents the number of pixels obtained after extracting the playing field from the j-th frame. h and w denote the height and width of the image, respectively, and $\frac{2}{3}$ indicates that only the bottom two-thirds of each frame is used.

3) Analyze the area of the playfield in key frames of medium shots and long shots to obtain a threshold for distinguishing these two groups of shots. If more than half of the frames in a shot are identified as long shots or medium shots, then the shot will be labeled with the corresponding tag; otherwise, the shot will be marked as a non-field shot and proceed to the next labeling phase in the decision tree.

After filtering out other types of shots, only external field and close-up shots remain. Close-up shots are directed towards the athletes, typically capturing their faces or upper bodies, with the background primarily being the competition field, as shown in Fig. 5.

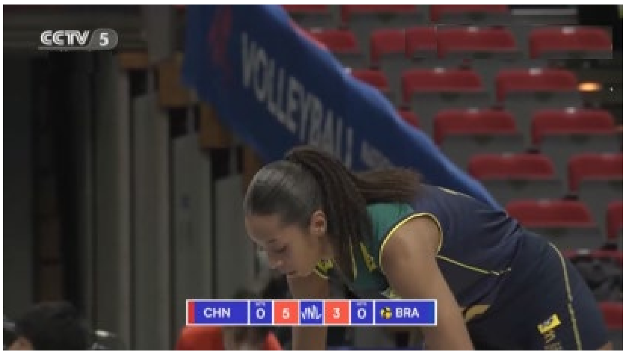


Fig. 5. Close-up shots in real match videos。

For external field shots, the main subjects in the frame are often multiple people with complex contours. Therefore, by using an algorithm to extract the edges of the image and calculate the proportion of edge pixels in the entire image, close-up shots can be distinguished from external field shots. The specific steps are as follows:

1) Apply the Canny edge detection algorithm to each image to identify the edge areas and calculate the proportion of edge pixels. The formula for this as Eq. (3):

$$ER(j) = \frac{D_2(j)}{w \times h} \tag{3}$$

where, $D_2(j)$ represents the number of edge pixels after edge detection for the j-th frame, and $ER(j)$ represents the edge pixel ratio of that image. The denominator $w \times h$ represents the total number of pixels of the image, which makes $ER(j)$ becomes a normalized ratio with a value between 0 and 1 that is used to fairly compare the texture complexity of images of different sizes.

2) Determine the threshold for the edge pixel ratio based on analysis and experimentation, and use it to identify key frames. If the edge pixel ratio is below this threshold, the frame is considered a close-up frame; otherwise, it is considered an external field frame.

3) Calculate the proportion of external field frames to close-up frames within a shot. If there are more external field frames, the shot is classified as an external field shot; conversely, if there are more close-up frames, it is classified as a close-up shot.

C. Human Object Detection

The key to player contour recognition is accurate and stable object detection. Single-stage object detection algorithms, as an efficient object detection framework, can simultaneously predict the category and location of objects in a single forward propagation, simplifying the object detection process and accelerating inference speed. The YOLO algorithm converts the object detection task into a regression problem by dividing the input image into a grid and predicting the bounding box and its class probability in each grid cell, achieving rapid detection. The advantage of this algorithm lies in its speed, which meets real-time requirements, but it has shortcomings in small object detection and image size adaptability. Therefore, based on the YOLOv8s algorithm, this paper uses the Feature Pyramid Network (FPN) structure to effectively avoid distortion and feature redundancy caused by image area operations, enhancing the ability to detect small objects. For player detection in videos, given the scarcity of volleyball domain datasets, data augmentation techniques are used to expand the dataset to address the issue of insufficient data.

The YOLOv8s object detection network mainly consists of four parts: the input end, the backbone network, the enhanced feature module, and the output end. Data augmentation methods are used during the data input phase, which helps to expand the dataset and prevent the model from overfitting. The backbone network is responsible for extracting image features. The enhanced feature module, through the Feature Pyramid Network (FPN), fuses high semantic information with low-level information to achieve multi-scale feature training, thereby improving the model's ability to detect small targets [38]. The output end is composed of decoupled modules that predict the targets and their corresponding bounding boxes.

Data augmentation Mosaic is a method to improve the quality of samples. Its operation involves randomly selecting four images from the training dataset, performing cropping, flipping, and other operations on the images, and then splicing and cropping the processed images back to their original size. This operation enriches the information of the target background. At this time, the Mixup method is also used, which randomly selects two images and fuses pixel values by direct interpolation. The computing equaitons as Eqs. (4) and (5):

$$x_n = \lambda x_i + (1 - \lambda) x_j \tag{4}$$

$$y_n = \lambda y_i + (1 - \lambda) y_j \tag{5}$$

where, (x_n, y_n) represents the fused pixel values, (x_i, x_j) and (y_i, y_j) represent the pixel values of the randomly selected images, and λ is a preset parameter. The parameter λ is usually sampled from a beta distribution and is used to control the degree of mixing. When $\lambda = 1$, the output is exactly equivalent to the first image, while when $\lambda = 0$, it is exactly equivalent to the second image. This operation forces the model to learn smoother decision boundaries through linear interpolation, which improves generalization.

Considering that features at different scales can provide richer information, and due to the fact that the size of players may vary under a single viewpoint, this paper proposes the use of convolutional kernels of different sizes in the deep feature extraction part of the model. Through a bottom-up feature extraction process, even if the input is of only one scale, a feature pyramid module with strong semantic features at all levels can be constructed.

In most traditional network architectures, object recognition and localization tasks are often designed to be executed in parallel on a single feature map. However, this design may not fully consider the essential differences in the requirements of the two tasks: the recognition task focuses more on identifying subtle differences between sample features, while the localization task pays more attention to the contour and shape features of the target object. Therefore, YOLOv8 adjusts the channels of the feature map through a decoupling module and then sends them to two different task branches. In these two branches, two 3×3 convolution operations are stacked separately to perform classification and regression tasks, respectively. The network structure is shown in Fig. 6. It is YOLOv8s decoupling module network.

Fig. 6 depicts the decoupled head structure implemented in the YOLOv8s architecture. In contrast to traditional coupled designs that rely on shared convolutional features for both classification and localization, this conFig.uration employs two dedicated, parallel branches. One branch specializes in classification tasks—specifically identifying objects as "person"—while the other focuses exclusively on regression tasks for precise bounding box coordinate prediction. This functional separation enables each branch to optimize its parameters for distinct feature representations, thereby enhancing localization precision and classification reliability compared to earlier integrated head designs.

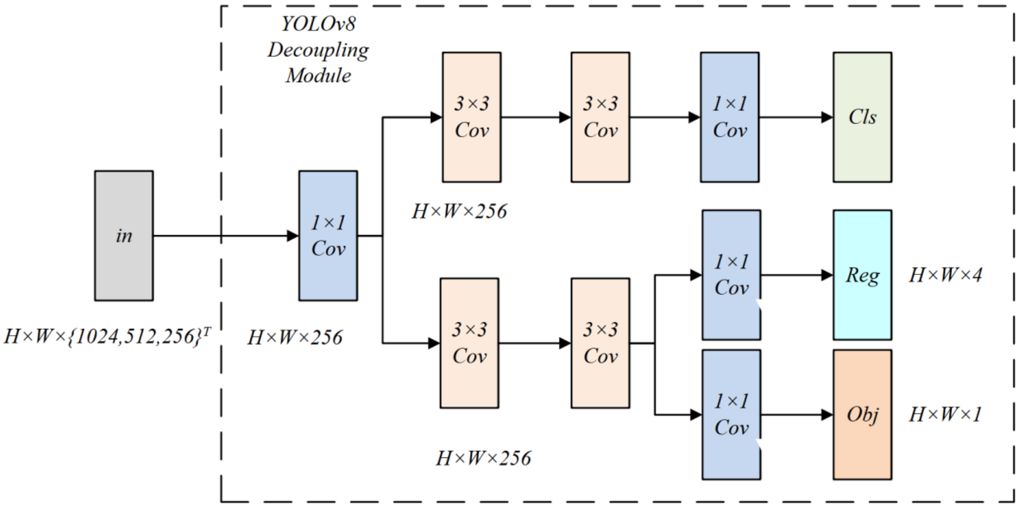


Fig. 6. YOLOv8s decoupling module network.

The improved YOLO network architecture is as follows:

1) Lightweight feature extraction network

The backbone network of YOLOv8 relies on the CSPDarknet for feature extraction, which has shown good performance in terms of object detection accuracy. However, due to its complex network structure and large number of parameters, the detection speed is affected. To enable the model to adapt to different computing environments, optimization becomes crucial. Currently, researchers have proposed various methods to improve the speed of deep learning models, including model lightweighting, merging BN layers, network pruning and quantization, and tensor decomposition techniques.

Regarding the merging of BN layers, it is commonly used during training to prevent overfitting and promote faster model convergence. However, this practice may increase the computational cost of the model, thereby slowing down the detection speed. Network pruning techniques remove weights that have minimal impact on model performance after training, especially those close to zero. For convolutions with zero weights, dilated convolutions can be applied as a solution. Given these considerations, this paper proposes an improved scheme to enhance detection efficiency by adopting a lightweight feature extraction network to optimize the model. GhostNet serves as this lightweight feature extraction network, relying minimally on 3×3 convolutions to construct the base layers and generating redundant feature layers based on efficient linear operations. This method significantly reduces the model's parameters while having a minimal impact on detection results, thereby enabling faster model operation.

2) Introducing an attention module

The Attention Mechanism (Attention Mechanism) originates from the way the human brain processes information [39]. It has been widely applied in Convolutional Neural Networks (CNNs) for both Natural Language Processing (NLP) and Computer Vision. This mechanism enhances the role of critical information and diminishes the impact of non-critical information by assigning different weights to various pieces of information, thereby improving the stability of the model. Depending on the level at which attention is applied, it can be categorized into three types: Channel Attention, Spatial Attention, and Hybrid Attention.

The Channel Attention Mechanism (Channel Attention Module, CAM) focuses on analyzing the interrelationships among different channels in a feature map and evaluates their importance by assigning different weights to each channel. The Spatial Attention Mechanism (Spatial Attention Module, SAM) emphasizes identifying key pixel regions within a feature map. However, this focus may sometimes cause the model to overlook non-critical information such as the background.

The Squeeze-and-Excitation (SE) network is a classic channel attention model, consisting of two parts: the Squeeze Block and the Excitation Block. The network structure is shown in Fig. 7 below.

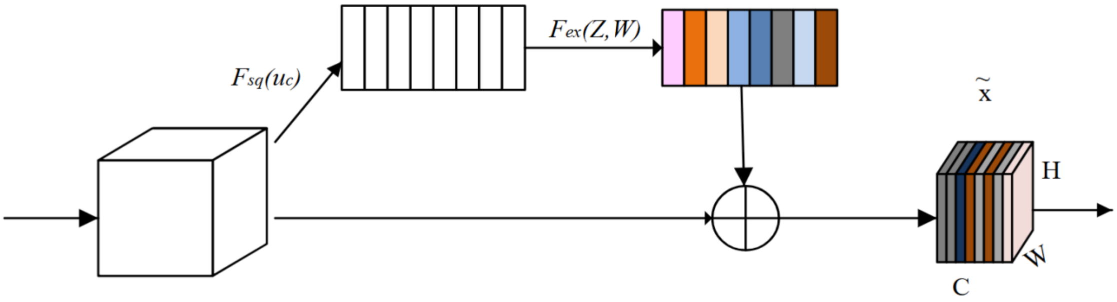


Fig. 7. SE network architecture.

The so-called Squeeze Block has a core function of compressing data containing multiple channels into a single-dimensional vector. Subsequently, matrix calculations are performed using trainable weights associated with this vector, with the aim of enhancing the saliency of features. These weights are adjusted through the backpropagation algorithm, enabling the model to identify and retain important features. The computational formula for the Squeeze Block as Eq. (6):

$$F_{sq}(u_c) = \frac{1}{HW}\sum_{i=1}^{H}\sum_{j=1}^{W} u_c(i,j) \tag{6}$$

where, u_c represents the feature of the c-th input matrix channel. The compressed vector is then fed into a neural network, using the ReLU function as the activation function. The activation of the features is completed through the Sigmoid function. Next, a scaling operation is performed, where the resulting output vector is multiplied element-wise with the original feature map to obtain a weighted feature map. This step not only enhances the important features but also weakens the impact of less important features, making the final extracted features more representative. The formula as Eq. (7):

$$\tilde{x_c} = F_{scale}(u_c, s_c) = s_c \times u_c \tag{7}$$

In the field of object detection, the size of the bounding box is crucial for the accuracy of object information. However, the SE network fails to fully address the issue of edge information loss during the feature compression process.

The CBAM network is composed of a spatial attention module and a channel attention module [40]. Its network structure is shown in Fig. 8. It is the structure of the CBAM module.

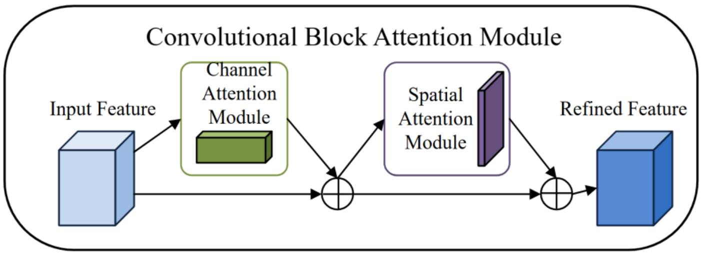


Fig. 8. The structure of the CBAM module.

Fig. 8 presents the detailed architecture of the Convolutional Block Attention Module (CBAM) integrated into our framework. This module progressively refines intermediate feature maps through a structured two-stage attention mechanism. Initially, the channel attention component evaluates the significance of individual feature channels, effectively determining "what" contextual information merits emphasis. Subsequently, the spatial attention component identifies salient regions within the feature maps, establishing "where" the model should concentrate its focus. This sequential processing enables the network to selectively amplify features associated with players while simultaneously suppressing non-essential background elements, thereby proving particularly advantageous for interpreting the complex visual environments characteristic of volleyball matches.

Under the framework of channel attention mechanism, global max pooling and average pooling operations are first performed on the feature map to reduce the dimensions in the width and height of the feature map. Subsequently, the processed feature map is fed into a multilayer perceptron network, which outputs two different feature vectors. The corresponding elements of these two vectors are added together and a Sigmoid function is applied to generate a weight vector for the channel features. Finally, this weight vector is used to multiply with each channel of the original input feature map, thereby enhancing the spatial attention of the feature map and obtaining a map that highlights important features more prominently. The expression as Eq. (8):

$$M_c(F) = \sigma(MLP(AvgPool(F) + MLP\big(MaxPool(F)\big)\big) \quad (8)$$

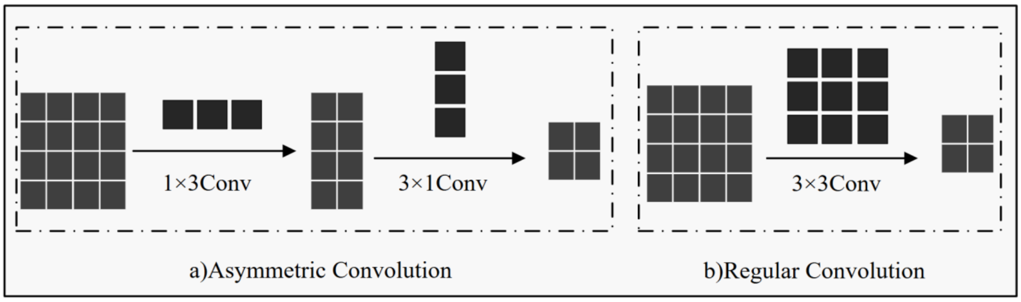
In the spatial attention mechanism, when processing feature maps, global max pooling and global average pooling techniques are employed to capture global information from the image. Subsequently, the results of these two pooling methods are concatenated along the spatial axis to integrate the information obtained from different pooling strategies. Then, a 7×7 convolutional layer is used to further refine the features, and a Sigmoid activation function is applied to determine the importance of the features. Ultimately, this process generates a feature map that integrates spatial and channel attention information, enabling the model to more accurately identify key features. The expression as Eq. (9):

$$M_c(F) = \sigma\big(f^{7\times7}([AvgPool(F); MaxPool(F)])\big) \quad (9)$$

Using asymmetric convolution, 3×3 dilated convolution is decomposed into two convolutions of 1×3 and 3×1 [41]. This operation retains only $\frac{2}{3}$ of the parameters of a regular convolution. The calculation of the compression ratio as Eq. (10):

$$q = \frac{m}{M} = \frac{N^2}{2N} \quad (10)$$

where, m represents the number of parameters in a regular convolution, while M represents the number of parameters in an asymmetric convolution. Taking a 4×4 input as an example, the computational comparison between the two types of convolutions is shown in Fig. 9. It is computational process comparison for asymmetric convolution as Fig. 9a) based on asymmetric mode 1×3:3×1 and regular convolution as Fig. 9b) based on regular window 3×3.



a)Asymmetric Convolution

b)Regular Convolution

Fig. 9. Computational process comparison for asymmetric convolution and regular convolution.

The network structure of the improved model, incorporating the aforementioned optimization techniques, is shown in Fig. 10.

Fig. 10 illustrates the complete architecture of our proposed Pro-YOLOv8s model, highlighting three primary modifications to the baseline YOLOv8s framework. The backbone network incorporates GhostNet to reduce computational complexity and parameter count through efficient linear operations. The enhanced features are subsequently processed by a Convolutional Block Attention Module (CBAM), which selectively emphasizes meaningful spatial and channel information. For bounding box regression, the conventional Complete Intersection over Union (CIoU) loss is replaced by the SIoU function to improve localization accuracy. Collectively, these optimizations yield a model that achieves superior detection performance while maintaining a reduced computational footprint, rendering it particularly suitable for deployment in real-time sports analytic applications.

The improved loss function is as follows: The calculation of the loss function is an essential part of object detection algorithms. The regression loss function for the prediction box in the YOLOv8s algorithm is CIoU, which takes into account the overlap area, the distance between the centers, and the aspect ratio of the predicted bounding box and the ground truth box. The specific formula for CIoU as Eqs. (11)–(13):

$$L_{CIoU} = IoU - \left(\frac{\rho^2(B^{pred},B^{gt})}{c^2} + \alpha v\right) \tag{11}$$

$$v = \frac{4}{\pi}\left(\tan^{-1}\frac{w^{gt}}{h^{gt}} - \tan^{-1}\frac{w^{pred}}{h^{pred}}\right) \tag{12}$$

$$\alpha = \frac{v}{(1-IoU)+v} \tag{13}$$

where, α is a balancing coefficient, v is used to measure the length and width ratio, and $\rho^2(B^{pred}, B^{gt})$ represents the Euclidean distance between the centers of the predicted box and the ground truth box.

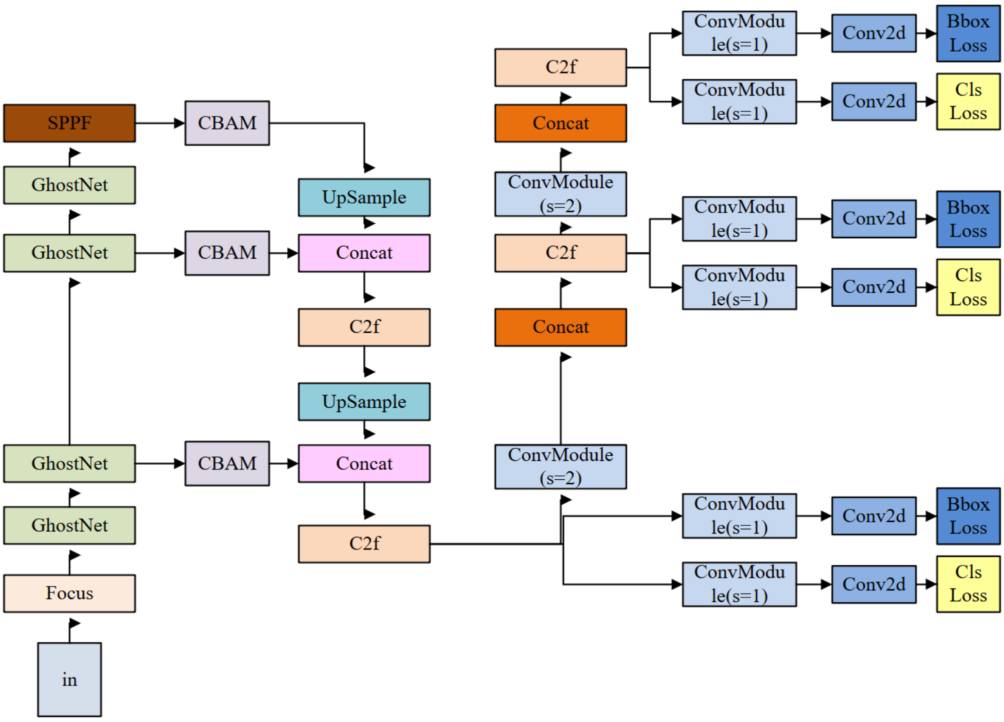


Fig. 10. The improved YOLOv8 network architecture.

However, the description of the aspect ratio in CIoU is somewhat subjective, which introduces a degree of uncertainty and ignores the directional differences between the ground truth box and the predicted box. Therefore, this paper proposes replacing CIoU with SIoU to improve training and convergence speed. The calculation of SIoU is divided into four parts: IoU loss, angle loss, distance loss, and shape loss. The formula for calculating the angle loss as Eq. (14):

$$\theta = cos\left(2 \times \left(sin^{-1}(sin\,\gamma) - \frac{\pi}{4}\right)\right) \tag{14}$$

where, $sin\,\gamma$ is the sine value of the angle between the centers of the predicted box and the ground truth box, calculated based on the differences in width and height of the centers. The formula for calculating the distance loss as Eq. (15):

$$P_x = 2 - exp(\beta \times \rho_x) - exp(\beta \times \rho_y) \tag{15}$$

where, ρ_x and ρ_y are the normalized values of the width and height differences between the centers, respectively, and β is the adjustment coefficient for the angle loss. The formula for calculating the shape loss as Eq. (16):

$$\Omega = (1 - exp(-\omega_w))^4 + (1 - exp(-\omega_h))^4 \tag{16}$$

where, ω_w and ω_h are the normalized differences in width and height between the predicted box and the ground truth box, respectively. IoU, as a scale-invariant metric, can measure the similarity between two rectangles of different shapes. The calculation formula for IoU as Eq. (17):

$$IoU_{A,B} = \frac{\|A \cap B\|}{\|A \cup B\|} = \frac{\|I\|}{\|U\|} \quad (17)$$

where, *A* and *B* are the boundary boxes of the predicted box and the ground truth box, respectively. Symbol $\|A \cap B\|$ calculates the intersection area between the predicted box and the true box, while $\|A \cap B\|$ computes the area of their union. As a scale invariant index, IoU has a range of [0, 1], 1 for perfect coincidence and 0 for perfect no overlap, which directly measures the accuracy of localization. *I* denotes the intersection, and *U* denotes the union.

Combining the four parts mentioned above, the expression for the SIoU loss function as Eq. (18):

$$L_{SIoU} = 1 - DIoU + \frac{P_x + \Omega}{2} \quad (18)$$

D. Experimental Setup

The experiments in this paper were conducted on the Ubuntu 22.04 operating system, using an NVIDIA RTX 4070 GPU. The environment includes CUDA 12.3, PyTorch 1.13.1, and Python 3.11 [44–48]. The YOLOv8s model was used as the baseline, following the default conFig.uration of v8. The hyper-parameters used during training are shown in Table I. The images were resized to 640×640 pixels, and data augmentation techniques such as Mosaic and Mixup were employed. The initial number of training iterations was set to 500, and after model lightweighting, the number of iterations was reduced to 100.

TABLE I. HYPER-PARAMETER SETTINGS

Parameter Name	Meaning	Default Value
learning_rate	Initial Learning Rate	0.01
momentum	Learning Rate Momentum	0.937
weight_decay	Weight Decay Coefficient	0.0005
epoch	Number of Training Epochs	500
lw_epoch	Number of Training Epochs after Model Lightweighting	100
batchsize	The number of samples used in one iteration	16

E. Experimental Dataset

Datasets are crucial for deep learning algorithms, not only ensuring fair comparisons between algorithms but also bringing new challenges to algorithm research through continuous expansion and improvement. The PASCAL VOC 2012 dataset is a publicly available dataset that provides a unified data format, high-quality images, and detailed annotations. It covers 20 categories, including people, animals, and vehicles, and contains approximately 11,000 images and 27,000 annotated objects. By writing a Python script to filter images labeled with “person,” we obtained a training set of 1994 images and a validation set of 2093 images. We then randomly selected 1006 images from the validation set to add to the training set, leaving the remaining 1087 images for validation [49–52].

The COCO (Common Objects in Context) dataset provides a large-scale, diverse, and practical benchmark for image recognition and object detection, containing 330,000 training images, 35,000 validation images, and 50,000 test images. By writing a Python script to filter images labeled with the “person” tag in the COCO dataset, we identified 64,115 images in the training set and 2693 images with corresponding labels in the validation set. These images will be used in this study.

F. Evaluation Metrics

The Volleyball dataset is a specialized dataset for volleyball match videos. It is the only publicly available dataset for multi-person action recognition and is currently the largest dataset for group activity recognition. The dataset consists of 55 volleyball match videos and 4,830 annotated frames [53–58]. Among these, 24 video sequences are used for training, 15 for validation, and 16 for testing. Each frame is annotated with bounding boxes and coordinates of the players, as well as nine individual action labels (waiting, passing, diving, falling, spiking, blocking, jumping, moving, and standing) and eight group activity labels (right pass, right spike, right reception, right score, left score, left pass, left spike, and left reception).

To evaluate the performance of player detection algorithms, appropriate evaluation metrics are necessary. The F1-score is a metric that takes into account both precision and recall. Average Precision (AP) is calculated by measuring the area under the Precision-Recall (PR) curve, typically using the 101-point interpolation method. This involves taking 101 points with a step size of 0.01 in the recall range from 0 to 1, and for each point, the precision value is the maximum value to its right. AP50 is the average precision calculated at an Intersection over Union (IoU) threshold of 0.5, meaning that a detection is considered correct if the overlap between the predicted box and the ground-truth box exceeds 50%. The mean Average Precision (mAP) is obtained by averaging the individual AP values and is also used for model evaluation. However, in this study, we focus primarily on the “person” category label, so its mAP value is essentially the AP value. The formulas for calculating precision, recall, and F1-score as Eqs. (19–(21):

$$Precision = \frac{TP}{TP + FP} \quad (19)$$

$$Recall = \frac{TP}{TP + FN} \quad (20)$$

$$F1 = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (21)$$

where, True Positive (*TP*) represents the number of samples that are predicted as positive by the model and are actually positive. False Positive (*FP*) is the number of samples that are predicted as positive by the model but are actually negative. False Negative (*FN*) is the number of samples that are predicted as negative by the model but are actually positive.

IV. Result and Discussion

A. Experimental Results and Comparative Analysis

The Precision-Recall (P-R) curve of the improved YOLOv8s shows high precision and recall, with an mAP value reaching 84.8%. The fullness of the curve indicates good robustness.

To evaluate the practical performance of the improved model, this paper compares the improved YOLOv8s with Faster R-CNN, YOLOv3, YOLOv5s, YOLOv7-tiny, and the original YOLOv8s models on the filtered datasets and analyzes their performance [42, 43]. The experimental results on the VOC dataset are shown in Table II, it is comparison of experimental performance on the PASCAL VOC 2012 Validation Set on our cared relative parameters. The results on the COCO dataset are shown in Table III, it is comparison of experimental performance on COCO Validation Set on our cared relative parameters. And the results on the Volleyball dataset are shown in Table IV, it is comparison of experimental performance on the Volleyball Validation Set on our cared relative parameters.

TABLE II. Comparison of Experimental Performance on the PASCAL VOC 2012 Validation Set

Method	GFLOPs	F1/%	mAP@50/%	FPS/s
Faster RCNN	180	80	78.8	39
YOLOv3	85	72	77.5	44
YOLOv5	53	84	81.5	76
YOLOv7-tiny	23	72	78.3	103
YOLOv8s	53	91	83.3	71
Pro-YOLOv8s	19	90	84.8	87

TABLE III. Comparison of Experimental Performance on COCO Validation Set

Method	GFLOPs	F1/%	mAP@50/%	FPS/s
Faster RCNN	180	68	58.7	37
YOLOv3	85	43	42.1	43
YOLOv5	53	90	66.7	80
YOLOv7-tiny	23	76	51.2	107
YOLOv8s	53	70	81.8	88
Pro-YOLOv8s	19	71	81.7	93

TABLE IV. Comparison of Experimental Performance on the Volleyball Validation Set

Method	GFLOPs	F1/%	mAP@50/%
Faster RCNN	180	71	77.5
YOLOv3	85	55	75.3
YOLOv5	53	73	82.5
YOLOv7-tiny	23	72	80.7
YOLOv8s	53	83	84.6
Pro-YOLOv8s	19	84	85.4

By analyzing the experimental results mentioned above, the optimized YOLOv8s model outperforms other models in terms of average precision and meets the requirement for real-time processing, achieving a balance between accuracy and speed. Compared with the original version, the optimized model reduces the number of parameters and computational load while maintaining a similar average precision and improving detection speed. Although its detection speed is slightly slower than that of YOLOv7-tiny, its accuracy is significantly higher, which meets the requirements for subsequent tasks.

B. Ablation Study

To verify whether the introduction of attention mechanisms improves performance, SE and CBAM attention modules were added respectively after the network's feature extraction, and the results were compared. The specific comparison results are shown in Table V, it is comparison of Experiments with Different Attention Mechanisms on our cared relative parameters, such as Precision, Recall, mAP@50/%.

TABLE V. Comparison of Experiments with Different Attention Mechanisms

Method	Precision/%	Recall/%	mAP@50/%
YOLOv8s	83.7	74.3	83.3
YOLOv8s+SE	85.4	75.6	83.0
YOLOv8s+CBAM	85.5	76.5	83.9

Compared with YOLOv8s, the model performance decreased after adding the SE module, which implies that the SE mechanism may not always be effective across different datasets. The CBAM attention mechanism, which focuses on both channel and spatial dimensions simultaneously, led to a 0.5 percentage point increase in the model's mAP and achieved the best performance in both detection precision and recall, demonstrating that the addition of CBAM had the most significant impact. Table VI shows the comparison results of the ablation study, where the first and second groups represent the effects of using the CBAM module and the SIoU loss function individually, while the third group presents the final detection model combining both. It is Ablation Study Results of the Optimized YOLOv8s on our cared relative parameters, such as Precision, Recall, mAP@50/%.

TABLE VI. Ablation Study Results of the Optimized YOLOv8s

Method	Module	Precision/%	Recall/%	mAP@50/%
YOLOv8s	-	83.7	74.3	83.3
1	CBAM	85.4	75.6	83.0
2	SIoU	84.8	76.7	83.7
3	CBAM+SIoU	85.5	76.5	83.9

As shown in Table VI, YOLOv8 uses the CIoU loss function. Group 1 involves adding the CBAM module but retaining the original CIoU loss function, while Group 2 uses the SIoU loss function without the CBAM module. Both groups have lower precision and recall compared to Group 3, which combines the CBAM module and the SIoU loss function in the optimized YOLOv8s.

The precision, recall, and mAP of the optimized YOLOv8s are improved by 1.7%, 3.8%, and 1.6%, respectively. This validates that the introduction of the CBAM module and the SIoU loss function positively impacts the model's performance, enabling better detection of human bodies in motion and thus providing a solid foundation for the subsequent tasks in this study. Table VI does NOT include the impact of these modifications on speed (FPS) and computational load (GFLOPs) because of these modifications are minor impact for speed (FPS) and computational load (GFLOPs), and are no need for discussion in our cared relative parameters.

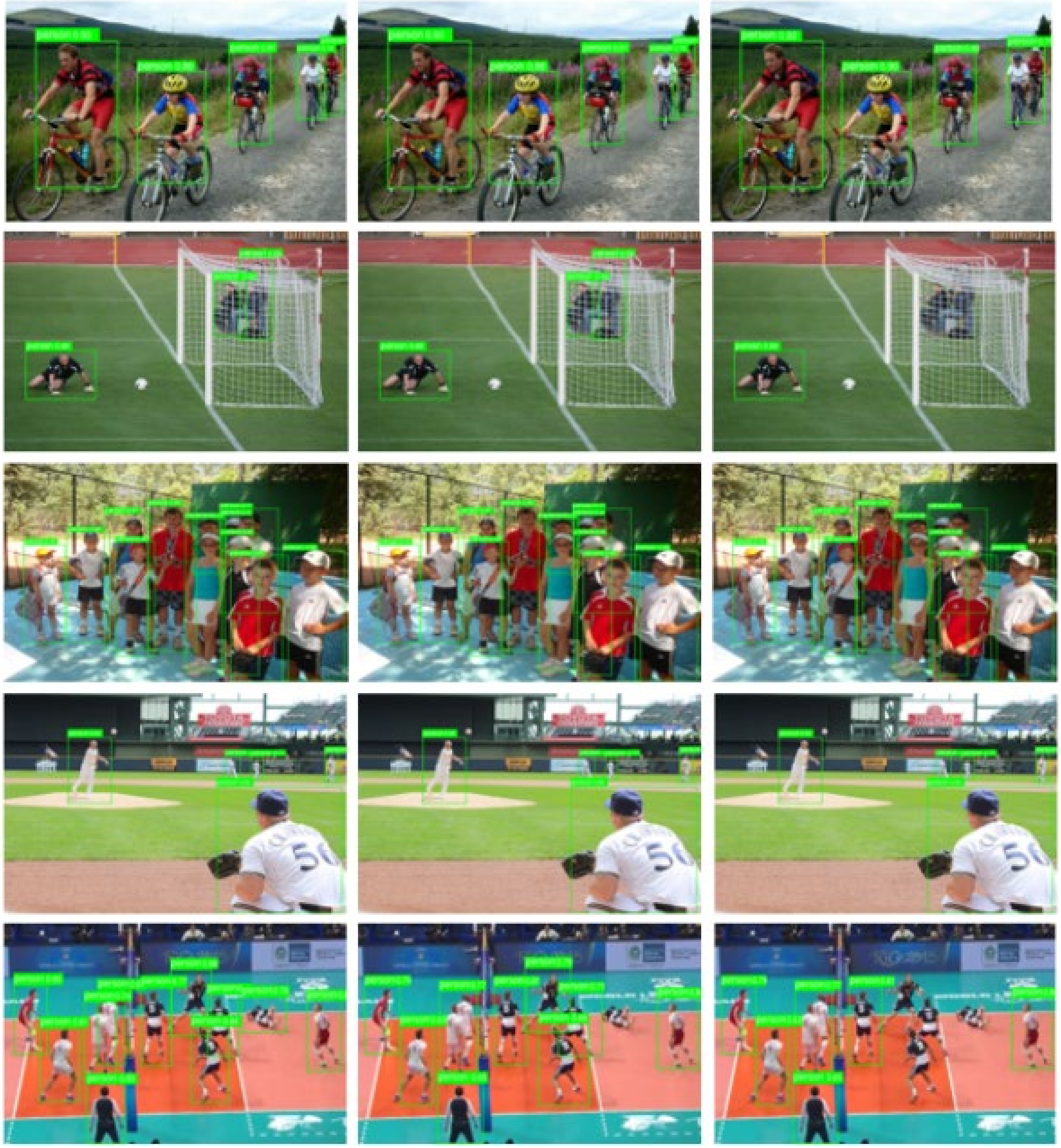


Fig. 11. Comparison of visualization effects on the VOC & COCO datasets.

To visually compare the optimized model with other models in human target detection on the same images, a visualization is presented in Fig. 11. The first row of images is from the PASCAL VOC 2012 training set, the second row from the PASCAL VOC 2012 validation set, the third row from the COCO training set, the fourth row from the COCO validation set, and the fifth and sixth rows are both from the Volleyball validation set. By examining Fig. 11, it can be observed that the optimized model Pro-YOLOv8s (as the 1rt column) maintains consistency with YOLOv8s (as the 2nd column) in detection performance. In scenes with multiple people, both outperform the YOLOv7-tiny (as the 3rd column) model, effectively reducing computational load while minimizing missed detection.

C. Experimental Results on Real Competition Videos

The video materials used in the experiment were taken from the live broadcasts of the 2023 FIVB Volleyball Women's World League. As an international high-level competition, the production quality ensures the professionalism and standardization of the videos. From the 128 preliminary matches, 16 matches featuring the Chinese women's volleyball team were selected, and four matches were randomly chosen for analysis. These included the matches between China and Brazil (Japan leg), China and Canada (Hong Kong leg), China and Korea (Korea leg), and China and Japan (Japan leg) When processing the videos, the audio tracks were first removed, and then non-essential information such as commercials was manually edited out to ensure that only the key parts were retained. Subsequently, a shot segmentation algorithm was applied to these selected videos, and the experimental results are shown in Table VII. It is experimental results of video shot segmentation on our cared relative parameters, such as Real Shot, Predicted Shot, Segmentation Time.

TABLE VII. EXPERIMENTAL RESULTS OF VIDEO SHOT SEGMENTATION

Match Name	Time/min	Real Shot	Predicted Shot	Segmentation Time/s
China-Brazil	27	202	200	162
China-Canada	29	219	215	163
China-Korea	30	223	213	180
China-Japan	25	189	185	145

Table VII summarizes the quantitative performance of the proposed shot segmentation algorithm across four actual match recordings. The results demonstrate a close correspondence between the number of predicted shots and the ground truth annotations across all test cases. For instance, in the China-Brazil match, the algorithm identified 200 shots against a manual annotation of 202. This high level of agreement confirms the segmentation reliability. Furthermore, the computational overhead remains practical for real-world deployment, with a 27-min video processed in 162 s, underscoring the feasibility of integrating this preprocessing stage into a complete analysis pipeline.

From the selected four volleyball matches, logo images were extracted and used as templates to compare the similarity with keyframes of potential logo shots. If the similarity exceeds 70%, the shot is determined to be a logo shot. The results are shown in Table VIII. It is the Results of Identifying Logo Shots. Table IX presents the semantic annotation status of the volleyball match videos. It is the Results of Identifying Logo Shots.

TABLE VIII. THE RESULTS OF IDENTIFYING LOGO SHOTS

Match Name	Badge Count	FP	FN	Recall/%	Accuracy/%
China-Brazil	38	0	0	100	96.1
China-Canada	46	0	5	89.1	95.8
China-Korea	42	0	1	97.6	96.9
China-Japan	40	0	3	92.5	95.4

TABLE IX. THE RESULTS OF IDENTIFYING LOGO SHOTS

Match Name	Total Shot	Replay Shot	Medium Shot	Long Shot	Close-up Shot	External Shot
China-Brazil	202	19	82	62	29	10
China-Canada	219	23	86	65	34	11
China-Korea	223	21	89	67	36	10
China-Japan	189	20	76	56	28	9

The data in the table shows that after detecting the logos and replay shots in four complete matches, the accuracy obtained is high, and the recall rate is also quite high, with almost no missed detection. This indicates that the semantic annotation of the relevant shots using the method proposed in this paper is truly effective.

On the validation set composed of real volleyball match videos, the improved algorithm proposed in this paper was compared with other algorithms, and the relevant experimental results are shown in Table X. It is the experimental performance comparison was conducted on real match videos.

TABLE X. THE EXPERIMENTAL PERFORMANCE COMPARISON WAS CONDUCTED ON REAL MATCH VIDEOS

Method	GFLOPs	F1/%	mAP@50/%	FPS/s
Faster RCNN	180	60	69.1	20
YOLOv3	85	49	65.7	27
YOLOv5	53	66	70.7	50
YOLOv7-tiny	23	63	68.9	69
YOLOv8s	53	78	77.4	55
Pro-YOLOv8s	19	78	78.2	60

Table X presents the detection performance of the proposed Pro-YOLOv8s framework when processing high-resolution broadcast footage under real-world conditions. Although a reduction in frame rate is observed across all models due to the substantial input dimensions, our approach achieves the highest detection accuracy with a mAP@50 of 78.2% while maintaining a processing speed of 60 FPS. Notably, the model accomplishes this performance with significantly lower computational demand, requiring only 19 GFLOPs.

Human target detection was performed on the same image using the optimized model and other models for comparison, and the results were visually displayed. By observing the comparison of the visual effects, it can be seen that the optimized model performs better in real match videos.

D. Discussion

Based on experimental results, we can know how the optimized model achieves a balance between accuracy, computational load, and processing speed during real match video analysis. Because through the analysis of the experimental results, it can be seen that when processing real match videos, the high resolution of the input images generally leads to a decrease in the processing speed of various models. But the improved model proposed in this paper still maintains a certain processing speed, and the accuracy remains high, which is sufficient to meet requirements of subsequent tasks. These results validate that the architectural optimizations successfully reconcile the competing objectives of precision and efficiency, fulfilling a core prerequisite for practical implementation in resource -constrained environments. In terms of video processing, this paper performs shot segmentation and key-frame extraction on videos with audio removed. Through a semantic annotation method, the video is described as a sequence of shots composed of long shots, medium shots, close-ups, replays, and off-court shots, laying the groundwork for the subsequent extraction of player position information from the video.

In order to clarify why these specific components are particularly suitable for the volleyball video scenario, our explanation is as follows: the proposed algorithm in this paper performs well on two object detection datasets of the volleyball video scenario. Compared with the original model, it has fewer parameters and lower computational load, and is suit for meeting the requirements of efficiency and accuracy in subsequent pose estimation of the volleyball video scenario.

While the rule-based and threshold-driven decision tree method is effective, its generalization capability may be limited, so its limitations is existing, such as how to build the rules based on knowledge databases, how to self-tuning threshold and so on. In the future work, we will continue to study relative machine learning method to deal with it.

V. Conclusion

The standard YOLOv8s network performs well in human detection. However, its large parameter count and high computational load hinder practical deployment. To address this, we implemented a lightweight design. Therefore, this paper achieves model lightweighting by integrating the Ghost network with the backbone network and introduces the CBAM module to enhance the semantic information of the lightweight model, thereby improving model accuracy. The proposed algorithm in this paper performs well on two object detection datasets. Compared with the original model, it has fewer parameters and lower computational load, and is capable of meeting the requirements for efficiency and accuracy in subsequent pose estimation. Ultimately, this work demonstrates the significant practical potential of optimized deep learning solutions in transforming volleyball match analysis, with immediate applications in coaching decision-support and live broadcast enhancement.

While the proposed framework demonstrates competitive performance, its current form suggests several meaningful avenues for future development. Building on the lightweight architecture and video understanding pipeline established in this study, subsequent research will pursue three key directions. First, we aim to incorporate multi-modal data streams by integrating visual analysis with physiological metrics—such as heart rate variability and electro- myography signals—alongside tactical match statistics. This integrated approach would enable a more comprehensive assessment of athlete performance and fatigue patterns. Second, to enhance the practical utility of the system, we plan to implement explainable AI techniques that provide transparent rationale for the model's outputs. Generating interpretable feedback is essential for fostering trust among coaches and sports analysts who rely on these systems for strategic decisions.

Complementing these technical focuses, a third direction involves developing a unified, real-time analysis platform that seamlessly integrates player detection, kinematic analysis, and tactical evaluation. Such an end-to- end system could offer immediate insights during both training sessions and competitive matches. The methodologies presented in this paper, (particularly the efficient model design and structured video parsing framework), provide a solid foundation for these future endeavors. Ultimately, this work not only offers a functional tool for volleyball analytics but also illustrates a viable approach for deploying optimized deep learning solutions in dynamic sports environments, contributing to the evolving landscape of sports intelligence systems.

Conflict of Interest

The authors declare no conflict of interest.

Author Contributions

Ting Zhang and Shuqi Niu conducted the research; Lu Chen, Chenhui Dou and Tao Liu analyzed the data; Degan Zhang wrote the paper; Shuqi Niu, Lu Chen, Tao Liu and Degan Zhang are co-first authors; all authors had approved the final version.

Funding

This research work is supported by Key Project of Tianjin Natural Science Foundation (Grant No.25JCZDJ C00420, No.18JCZDJC96800).

References

[1] D. G. Zhang, J. X. Wang *et al.,* "A new method of fuzzy mutlicriteria routing in vehicle ad-hoc network," *IEEE Transactions on Computational Social Systems*, vol. 10, no. 6, pp. 3181–3193, 2022.

[2] J. Zhang, Z. H. Zhang *et al.*, "Novel approach of vehicular cooperative communication based on strategy of interval type-2 fuzzy logic and cooperative game," *IEEE Transactions on Sustainable Computing*, vol. 10, no. 3, pp. 588–600, 2025. doi: 10.1109/TSUSC.2024. 3503580

[3] J. Zhang *et al.,* "New offloading method of computing task based on gray wolf hunting optimization mechanism for the IOV," *IEEE Transactions on Network and Service Management*, vol. 22, no. 3, pp. 2264–2277, 2025. doi:10.1109/ TNSM.2025.3539865

[4] D. G. Zhang, G. Li *et al.*, "An energy-balanced routing method based on forward-aware factor for wireless sensor network," *IEEE Transactions on Industrial Informatics*, vol. 10, no. 1, pp. 766–773, 2014.

[5] D. Zhang, X. Wang, X. Song *et al.,* "A novel approach to mapped correlation of ID for RFID anti-collision," *IEEE Transactions on Services Computing*, vol. 7, no. 4, pp. 741–748, 2014.

[6] D. G. Zhang, W. M. Dong, T. Zhang *et al.*, "New computing tasks offloading method for MEC based on prospect theory framework," *IEEE Transactions on Computational Social Systems*, vol. 11, no. 1, pp. 770–781, 2024.

[7] G. Lavee, E. Rivlin *et al.*, "Understanding video events: A survey of methods for automatic interpretation of semantic occurrence in video," *IEEE Transactions on Systems Man & Cybernetics Part C Applications and Reviews*, vol. 39, no. 5, pp. 489–504, 2009.

[8] D. Zhang *et al.*, "New medical image fusion approach with coding based on SCD in wireless sensor network," *Journal of Electrical Engineering & Technology*, vol. 10, no. 6, pp. 2384–2392, 2015.

[9] D. W. Tjondronegoro and Y. P. Chen, "Knowledge-discounted event detection in sports video," *IEEE Transactions on System Men & Cybernetics Part A Systems & Humans*, vol. 40, no. 5, 1009–1024, 2010.

[10] M. Andriluka *et al.*, "Pictorial structures revisited: People detection and articulated pose estimation," in *Proc. 2009 IEEE Conference on Computer Vision and Pattern Recognition*, 2009, pp. 1014–1021.

[11] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Advances in Neural Information Processing Systems*, vol. 25, 2012.

[12] D. Zhang, H. Ge *et al.*, "New multi-hop clustering algorithm for vehicular ad hoc networks," *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 4, pp. 1517–1530, 2019.

[13] D. G. Zhang, H. Z. An *et al.*, "Novel privacy awareness task offloading approach based on privacy entropy," *IEEE Transactions on Network and Service Management*, vol. 21, no. 3, pp. 3598–3608, 2024.

[14] J. Tompson, R. Goroshin *et al.*, "Efficient object localization using convolutional networks," in *Proc. the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 648–656.

[15] D. G. Zhang, W. J. Wang *et al.*, "Novel edge caching approach based on multi-agent deep reinforcement learning for internet of

vehicles," *IEEE Transactions on Intelligent Transportation Systems*, vol. 24, no. 8, pp. 8324–8338, 2023.

[16] R. Girshick, J. Donahue, T. Darrell *et al.*, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proc. the IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 580–587.

[17] S. Ren, K. He *et al.*, "Faster R-CNN: Towards real-time object detection with region proposal networks," *IEEE Transactions on Pattern Analysis & Machine Intelligence*, vol. 39, no. 6, pp. 1137–1149, 2017.

[18] D. G. Zhang and S. Zhou., "A low duty cycle efficient MAC protocol based on self-adaption and predictive strategy," *Mobile Networks & Applications*, vol. 23, no. 4, pp. 828–839, 2018.

[19] D. Zhang *et al.*, "UAV-assisted task offloading system using dung beetle optimization algorithm & deep reinforcement learning," *Ad Hoc Networks*, vol. 156, 103434, 2024. doi: 10.1016/j.adhoc.2024.103434

[20] W. Liu, D. Anguelov *et al.*, "SSD: Single shot multibox detector," in *Proc. European Conference on Computer Vision*, 2016, pp. 21–37.

[21] J. Redmon and A. Farhadi, "YOLO9000: Better, faster, stronger," in *Proc. the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 7263–7271.

[22] J. Zhang, L. Zhang *et al.*, "New routing method based on sticky bacteria algorithm and link stability for VANET," *Ad Hoc Networks*, vol. 166, 103682, 2025. doi: 10.1016/j.adhoc.2024.103682

[23] D. G. Zhang, C. H. Ni *et al.*, "New method of vehicle cooperative communication based on fuzzy logic and signal game strategy," *Future Generation Computer Systems*, vol. 142, pp. 131–149, 2023.

[24] R. Varghese and M. Sambath, "YOLOv8: A novel object detection algorithm with enhanced performance and robustness," in *Proc. 2024 International Conference on Advances in Data Engineering and Intelligent Computing Systems*, 2024, pp. 1–6. doi:10.1109/ADICS58448.2024.10533619

[25] F. J. Du and S. J. Jiao, "Improvement of lightweight convolutional neural network model based on YOLO algorithm and its research in pavement defect detection," *Sensors*, vol. 22, no. 9, 3537, 2022.

[26] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in *Proc. the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 1251–1258.

[27] D. Zhang, Y. Cui, and T. Zhang, "New Quantum-Genetic Based OLSR Protocol (QG-OLSR) for mobile ad hoc network," *Applied Soft Computing*, vol. 80, pp. 285–296, 2019.

[28] M. Sandler, A. Howard *et al.*, "Mobilenetv2: Inverted residuals and linear bottlenecks," in *Proc. the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 4510–4520.

[29] A. Howard, M. Sandler, G. Chu *et al.*, "Searching for mobilenetv3," in *Proc. the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 1314–1324.

[30] X. Zhang, X. Zhou, M. Lin *et al.*, "Shufflenet: An extremely efficient convolutional neural network for mobile devices," in *Proc. the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 6848–6856.

[31] N. Ma, X. Zhang, H. T. Zheng *et al.*, "Shufflenet v2: Practical guidelines for efficient CNN architecture design," in *Proc. the European Conference on Computer Vision*, 2018, pp. 116–131.

[32] K. Han, Y. Wang, Q. Tian *et al.*, "GhostNet: More features from cheap operations," in *Proc. the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 1580–1589.

[33] S. Wenyu, C. Jian, L. Pu *et al.*, "Pruning and fine-tuning optimization method for convolutional neural network models using global information," *Acta Scientiarum Naturalium Universitatis Pekinensis*, vol. 57, no. 4, pp. 790–794, 2021.

[34] F. Jia, X. Wang, J. Guan *et al.*, "WRGPruner: A new model pruning solution for tiny salient object detection," *Image and Vision Computing*, vol. 109, 104143, 2021.

[35] D. Blalock, J. J. G. Ortiz, J. Frankle *et al.*, "What is the state of neural network pruning?" in *Proc. Machine Learning and Systems*, 2020, vol. 2, pp. 129–146.

[36] T. Liang *et al.*, "Pruning and quantization for deep neural network acceleration: A survey," *Neurocomputing*, vol. 461, pp. 370–403, 2021.

[37] D. Zhang, X. Wang, X. D. Song *et al.*, "A new clustering routing method based on PECE for WSN," *EURASIP Journal on Wireless Communications and Networking*, vol. 2015, no. 1, 162, 2015. doi:10.1186/s13638-015- 0399-x

[38] D. Zhang and C. Chen, "New method of energy efficient subcarrier allocation based on evolutionary game theory," *Mobile Networks and Applications*, vol. 26, no. 2, pp. 523–536, 2021.

[39] D. Zhang, Y. N. Zhu *et al.*, "A new constructing approach for a weighted topology of wireless sensor networks based on local-world theory for the Internet of Things (IoT)," *Computers & Mathematics with Applications*, vol. 64, no.5, pp. 1044–1055, 2012.

[40] G. D. Fu, J. Huang, T. Yang *et al.*, "Improved lightweight attention model based on CBAM," *Computer Engineering and Applications*, vol. 57, pp. 150–156, 2021

[41] D. Zhang, "A new approach and system for attentive mobile learning based on seamless migration," *Applied Intelligence*, vol. 36, no. 1, pp. 75–89, 2012.

[42] D. Zhang, J. Zhang, C. Ni *et al.*, "New method of edge computing-based data adaptive return in internet of vehicles," *IEEE Transactions on Industrial Informatics*, vol. 20, no. 2, pp. 2042–2052, 2024.

[43] C. Y. Wang, A. Bochkovskiy, and H. Y. M. Liao, "YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors," in *Proc. the IEEE Conference on Computer Vision and Pattern Recognition*, 2023, pp. 7464–7475.

[44] D. Zhang, C. Gong, T. Zhang *et al.*, "A new algorithm of clustering AODV based on edge computing strategy in IOV," *Wireless Networks*, vol. 27, no. 4, pp. 2891–2908, 2021.

[45] D. Zhang, S. Liu, T. Zhang *et al.*, "Novel unequal clustering routing protocol considering energy balancing based on network partition & distance for mobile education," *Journal of Network and Computer Applications*, vol. 88, no. 15, pp. 1–9, 2017. doi: 10.1016/j. jnca.2017.03.025

[46] Z. Ma *et al.*, "Shadow detection of moving objects based on multisource information in internet of things," *Journal of Experimental & Theoretical Artificial Intelligence*, vol. 29, no. 3, pp. 649–661, 2017.

[47] D. Zhang, M. Piao *et al.*, "New algorithm of multi-strategy channel allocation for edge computing," *AEU-International Journal of Electronics and Communications*, vol. 126, 153371, 2020. doi: 10.1016/j.aeue.2020.153372

[48] T. Zhang *et al.*, "A new method of data missing estimation with FNN-based tensor heterogeneous ensemble learning for internet of vehicle," *Neurocomputing*, vol. 420, pp. 98–110, 2021.

[49] D. Zhang, J. Gao *et al.*, "Novel approach of distributed & adaptive trust metrics for MANET," *Wireless Networks*, vol. 25, no. 6, pp. 3587–3603, 2019.

[50] J. Zhang, M. Piao *et al.*, "An approach of multi-objective computing task offloading scheduling based NSGS for IOV in 5G," *Cluster Computing*, vol. 25, no. 6, pp. 4203–4219, 2022.

[51] D. Zhang, T. Zhang *et al.*, "Novel self-adaptive routing service algorithm for application of VANET," *Applied Intelligence*, vol. 49, no. 5, pp. 1866–1879, 2019.

[52] J. Yang, M. Ding *et al.*, "Optimal base station antenna downtilt in downlink cellular networks," *IEEE Transactions on Wireless Communications*, vol. 18, no. 3, pp. 1779–1791, 2019. doi:10.1109/TWC.2019.2897296

[53] J. Chen, G. Mao *et al.*, "Capacity of cooperative vehicular networks with infrastructure support: Multi-user case," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 2, pp. 1546–1560, 2018. doi:10.1109/TVT.2017.2753772

[54] D. Zhang, W. Wang *et al.*, "New method of vehicular network content distribution based on edge caching and catch fish optimization strategy," *IEEE Transactions on Reliability*, 2025. doi: 10.1109/TR.2025.3617340

[55] D. G. Zhang, J. Zhang *et al.*, "Novel approach of computational resource allocation in fog computing based on deep reinforcement learning strategies," *IEEE Internet of Things Journal*, vol. 12, no. 20, pp. 43829–443841, 2025. doi: 10.1109/JIOT.2025.3598760

[56] A. J. Moshayedi, Z. Wang *et al.*, "Smart farming solutions: A user-friendly GUI for maize tassel estimation using YOLO with dynamic and fixed labelling, featuring video support," *IEEE Access*, 2025. doi: 10.1109/ACCESS.2025.3554984

[57] A. J. Moshayedi, A. S. Khan, K. Geng *et al.*, "Advancing agricultural practices: Analyzing the role of robotics in corn farming," *International Journal of Engineering*, vol. 38, pp. 1517–1532, 2025. doi: 10.5829/ije.2025.38.07a.07

[58] A. J. Moshayedi, A. S. Khan *et al*., "E-nose-driven advancements in ammonia gas detection: A comprehensive review from traditional to cutting-edge systems in indoor to outdoor agriculture," *Sustainability*, vol. 15, no. 15, 11601, 2023. doi: 10.3390/su151511601