Chen G and Liu S. (2024) CONSTRUCTING A BASIC TRAINING SYSTEM FOR FOOTBALL IN UNIVERSITIES BASED ON OBJECT DETECTION AND TRACKING ALGORITHMS. Revista Internacional de Medicina y Ciencias de la Actividad Física y el Deporte vol. 24 (96) pp. 120-145
DOI: https://doi.org/10.15366/rimcafd2024.96.008

ORIGINAL

CONSTRUCTING A BASIC TRAINING SYSTEM FOR FOOTBALL IN UNIVERSITIES BASED ON OBJECT DETECTION AND TRACKING ALGORITHMS.

Guangqing Chen[1], Shaoyong Liu[2*]

[1] Zhejiang Industry Polytechnic College, Shaoxing, Zhejiang,312000, China.
[2] Teacher Education College, Shaoxing University, Shaoxing, Zhejiang, 312000, China.
E-mail: liushaoyong2022@163.com

Recibido 07 de octubre de 2023 **Received** October 07, 2023
Aceptado 06 de puede de 2024 **Accepted** May 06, 2024

ABSTRACT

Football education plays a significant role in university education, enriching campus life and promoting holistic student development. However, university football training in China faces challenges due to limitations in popularity, training resources, and environmental conditions. This study addresses issues of decreased detection accuracy caused by occlusion, uneven lighting, and small object size in traditional object detection systems by improving the YOLOv5 network structure. The original CSPDarkNet53 backbone was streamlined to a Mobile Net structure with depth wise separable convolutions, reducing model parameters and improving detection speed. Attention mechanisms were integrated into the YOLOv5 network to enhance target feature extraction and mitigate background interference, addressing occlusion and complex backgrounds. Additionally, the Unscented Kalman Filter was introduced into Deep SORT, replacing IoU with DIoU and employing a cascade matching method, significantly reducing ID switching in target tracking tasks. Experimental results on public datasets demonstrate that the proposed model exhibits superior detection performance, making it suitable for university football training scenarios. This study also highlights the potential of artificial intelligence, particularly object detection technology, to advance the efficiency and effectiveness of football training, contributing to the comprehensive development of student-athletes.

KEYWORDS: University football training; object detection; real-time tracking; YOLOv5

1. INTRODUCTION

Football education plays a significant role in university education, enriching campus life and profoundly impacting students' holistic development and the overall quality of higher education. As a team sport, football requires close cooperation among players, fostering students' teamwork spirit and collective consciousness. Football activities and matches can enrich campus life, enhance campus vitality and cohesion, and promote interaction and communication between teachers and students. Through football training, students can achieve effective development in moral, intellectual, physical, and aesthetic aspects, forming a well-rounded personality and good qualities. Football education is of great importance in university education, encompassing aspects such as enhancing physical fitness and health, cultivating team spirit, improving psychological quality, enriching campus culture, providing diverse development opportunities, promoting comprehensive quality education, and fostering leadership and decision-making skills. Through football education, universities can achieve the goal of holistic education, helping students achieve comprehensive development in moral, intellectual, physical, and aesthetic aspects, and cultivating more well-rounded and outstanding talents for society.

In recent years, university football training in China has gradually gained attention and achieved some development. However, there are still challenges and limitations in terms of overall level and popularity. Under the guidance of national policies, an increasing number of universities have offered football elective courses or established on-campus football clubs, enhancing students' enthusiasm for participating in football. Various levels of university football leagues, such as the Chinese University Football League (CUFL), have gradually been carried out, providing students with more opportunities for competition and exchange platforms. Some universities actively promote football culture construction by organizing football cultural festivals and on-campus matches to create a strong football atmosphere.

Despite significant progress, university football in China also faces a series of challenges and limitations. The popularity and training level of university football are relatively high in first-tier cities and economically developed regions, while lower in remote areas and economically underdeveloped regions. The training and continuing education mechanisms for university football coaches are still imperfect, with some coaches lacking advanced training concepts and methods. Although some universities have hired professional football coaches, there is still an overall shortage of coaching resources and uneven professional levels. Some universities' inadequate investment in football training equipment and technological auxiliary devices affects the quality and effectiveness of training. Although the influence of football in universities is gradually increasing, the overall student participation

rate still needs improvement, especially among ordinary students and female students (Adham et al., 2022).

Artificial intelligence (AI) technology, especially object detection technology, has broad application prospects and significant importance in university football training in China. It can enhance the scientificity, efficiency, and effectiveness of training, promoting the intelligent and modernized development of university football training. Detailed training data collected through object detection technology can provide scientific basis for coaches to formulate more personalized and targeted training plans. Object detection technology can monitor players' positions and running routes on the field in real-time, analyze their tactical execution, and assist coaches in making tactical adjustments and optimizations. It can also monitor players' exercise load and fatigue levels in real-time, avoid overtraining, reasonably arrange training intensity, and reduce the risk of sports injuries. Intelligent coaching systems based on object detection technology can assist coaches in training guidance, providing scientific movement correction suggestions and training programs. AI, particularly object detection technology, has significant importance in university football training in China. It not only improves training efficiency and effectiveness but also promotes the development of tactical analysis, physical management, intelligent teaching, scientific research support, and campus football culture. Through the application of these technologies, university football training will become more scientific and intelligent, comprehensively enhancing players' overall quality and competitive level, and cultivating more outstanding talents for the development of Chinese football.

YOLO (You Only Look Once) is an advanced real-time object detection algorithm. It has widespread applications in the field of computer vision, especially in scenarios requiring real-time performance and high efficiency. Yang et al. proposed an improved object detection algorithm based on Dynamic Deformable Convolutional Networks (D-DCN) (J. Yang & Gapar, 2024), which enhances detection accuracy and robustness by introducing multi-scale feature fusion and dynamic offset calculation mechanisms. Experiments have verified significant performance improvements on the KITTI and Caltech datasets, demonstrating high practical value. Yang et al. proposed a lightweight object detection model YOLOv8-Lite based on the YOLOv8 framework (M. Yang & Fan, 2024), which improves model performance and efficiency through the introduction of the Fast Det structure, TFPN pyramid structure, and CBAM attention mechanism. Experimental results show that this model exhibits higher accuracy and robustness on the NEXET and KITTI datasets, making it suitable for intelligent transportation fields such as autonomous driving. Huang et al. proposed an improved YOLOv3 method for the detection of immature apples in orchard scenes (Huang, Zhang, Liu, & Li, 2023). By using the CSPDarknet53 backbone network, CIOU target box regression mechanism, and Mosaic algorithm, detection accuracy was improved. On a severely occluded dataset,

the method achieved an F1 score of 0.652 and a mAP of 0.675, with an inference speed of 12 milliseconds per image and a detection speed of 83 frames per second, showing significant detection effects. Jin et al. proposed the improved YOLOv7-bw algorithm for efficient remote sensing image detection (X. Jin et al., 2024). By introducing a dual-stage routing attention module and dynamic non-monotonic WIoUv3 loss function, feature extraction and detection effects were enhanced. On the DIOR dataset, YOLOv7-bw achieved mAP@0.5 and mAP@0.5:0.95 scores of 85.63% and 65.93%, respectively, improving by 1.93% and 2.03% compared to existing methods (Yasaman & Caitlin, 2023).

The application of YOLO in sports has significant importance. It can enhance various aspects such as tactical analysis and optimization, sports performance evaluation, physical monitoring and management, referee assistance and match fairness, and audience experience. Through the application of YOLO technology, sports training and competitions will become more scientific and intelligent, providing strong support for the comprehensive development and competitive level enhancement of athletes. Guntuboina et al. proposed a method for extracting and summarizing key events in sports videos based on YOLO and OCR (Guntuboina, Porwal, Jain, & Shingrakhia, 2021). By detecting scoreboards with YOLO and performing image processing and OCR recognition, timestamps of key events are generated. This method achieved an average F1 score of 0.979 in multiple sports video tests, suitable for precise match analysis. Zhang et al. proposed a real-time athlete tracking system based on YOLOv4 and Deep Sort for NBA and World Cup scenarios (Zhang, Chen, & Wei, 2020). The system achieved real-time tracking of each athlete, providing trajectory information and could be used for teaching and match reviews. Dwijayanto et al. utilized a YOLO deep neural network system for real-time detection of football field landmarks (Dwijayanto, Kurniawan, & Sugandi, 2019). The system ran at 16 frames per second on 608×608 pixel images, achieving a 97.60% MAP and 81.36% IoU accuracy. Diwan et al. proposed a neural network model for football training analysis, capable of real-time tracking of players and football (Diwan, Bandi, Dicholkar, & Khadse, 2023). The model is suitable for videos of any size, length, and quality, outputting bounding boxes with indices or identities for players and football. This is an important step towards real-time automated analysis of football training. Rangappa et al. introduced methods for training and customizing deep learning models for detecting, tracking, and identifying players in football training (Rangappa, Li, & Qian, 2021). By customizing camera settings and developing new spatial feature filters and bounding box position filters, players and spectators were distinguished. A novel method was proposed for player identification and tracking by detecting player jersey numbers with high confidence. Finally, a unique result evaluation technique was provided to assess model performance.

Although YOLO has achieved remarkable results in sports applications, there is still a gap in specific research and application in university football

training. Current research mainly focuses on applications in professional match scenarios, lacking customized research for university football training scenarios. University football training has unique needs and characteristics, such as the diversity of training fields and the varying technical levels of student-athletes, requiring specialized YOLO model optimizations. Existing studies mainly focus on object detection and tracking in matches, with less research on data collection and analysis in university football training. How to use YOLO technology to collect training data in real-time, analyze student-athletes' performance, and provide targeted training improvement suggestions remains an unsolved issue. This paper focuses on improving the YOLOv5 network structure for university football training video scenarios with the following innovations:

1. To address the problem of large network parameters and long detection times caused by the complex backbone structure of the network, which cannot meet real-time detection requirements, this paper light weighted the YOLOv5 backbone network, reducing the number of parameters and improving detection speed.

2. To overcome the interference caused by the football field background and the occlusion phenomenon among targets, the attention mechanism was integrated into the YOLOv5 network structure, enhancing the extraction of target features and weakening the interference of irrelevant backgrounds, effectively solving the problems of complex background and target occlusion affecting detection accuracy.

3. This paper introduced the Unscented Kalman Filter into DeepSORT, replaced IoU with DIoU, and combined it with a cascade matching method to match target detection boxes with appearance models. Experimental results show that the improved DeepSORT reduces the occurrence of ID switching in target tracking tasks while maintaining high detection speed.

2. Target Detection Algorithm for Basic Football Training Based on YOLOv5

In practical football training scenarios, the accuracy of target detection is often affected by factors such as dense distribution of targets, occlusion between targets, and small target sizes, leading to a decrease in detection precision. Therefore, accurately and quickly detecting targets on the football field in complex environments has become a current challenge in object detection.

Based on the original YOLOv5 network model, this paper addresses the issues of dense targets, occlusion, and small target sizes in football training scenarios that cause performance degradation in the original YOLOv5 network. We optimized the original YOLOv5 network structure by streamlining its

backbone network, modifying the neck part of the original YOLOv5 structure, and embedding the attention mechanism into the network structure. Finally, the neural network was trained on public datasets. Experimental results show that the modified YOLOv5 has enhanced detection capabilities (Wang, Zhang, Cheng, & Al-Nabhan, 2021; Zhou, Chen, & Xu, 2022).

2.1 Original YOLOv5 Model

Through the analysis of multi-target detection in football training scenarios, we identified several issues such as complex backgrounds interfering with detection, occlusion between targets, and small target sizes leading to missed or incorrect detections. First, frames from the football training scene are loaded sequentially, and each loaded frame is preprocessed into an RGB image for target detection using the YOLOv5 model. The final output consists of labeled bounding boxes with classifications and confidence scores for the detected targets.

The most critical part of the target detection process is the construction of the detection network, as the quality of the network structure directly impacts detection performance. In this study, the original YOLOv5 detection model was chosen as the baseline standard for comparison with the improved model. As previously discussed, the baseline YOLOv5 model comprises three main components: the backbone network, the neck, and the head. The backbone network uses CSPDarknet-53, the neck part employs a combination of PANet and SPP structures, and the head includes three convolutional layers that output the bounding box positions, confidence scores, and classes, respectively. The network structure of the baseline YOLOv5 is illustrated in Figure 1.

As shown in the figure, the YOLOv5 network accepts input images of size 640 × 640 with three color channels (RGB). After input, the images are processed by the backbone network for feature extraction, which involves generating feature vectors using convolutional kernels and continuously down-sampling to obtain feature maps of different sizes and channels: 80 × 80 × 255, 40 × 40 × 255, and 20 × 20 × 255. The backbone structure mainly consists of Conv modules and C3 modules. The Conv module represents a cascade combination of convolution operations (conv), batch normalization processing (BN), and the SiLU activation function. The SiLU function is expressed as:

$$SiLU(x) = x \cdot \sigma(x) = x \cdot \frac{1}{1+e^{-x}} \qquad (1)$$

Compared to the ReLU function, the SiLU function is smoother and is not monotonically increasing within its domain. Additionally, when the input is less than zero, the function graph is not horizontal and still possesses a gradient. The minimum value of the function acts as a soft floor for the weights, inhibiting the learning of large weights.

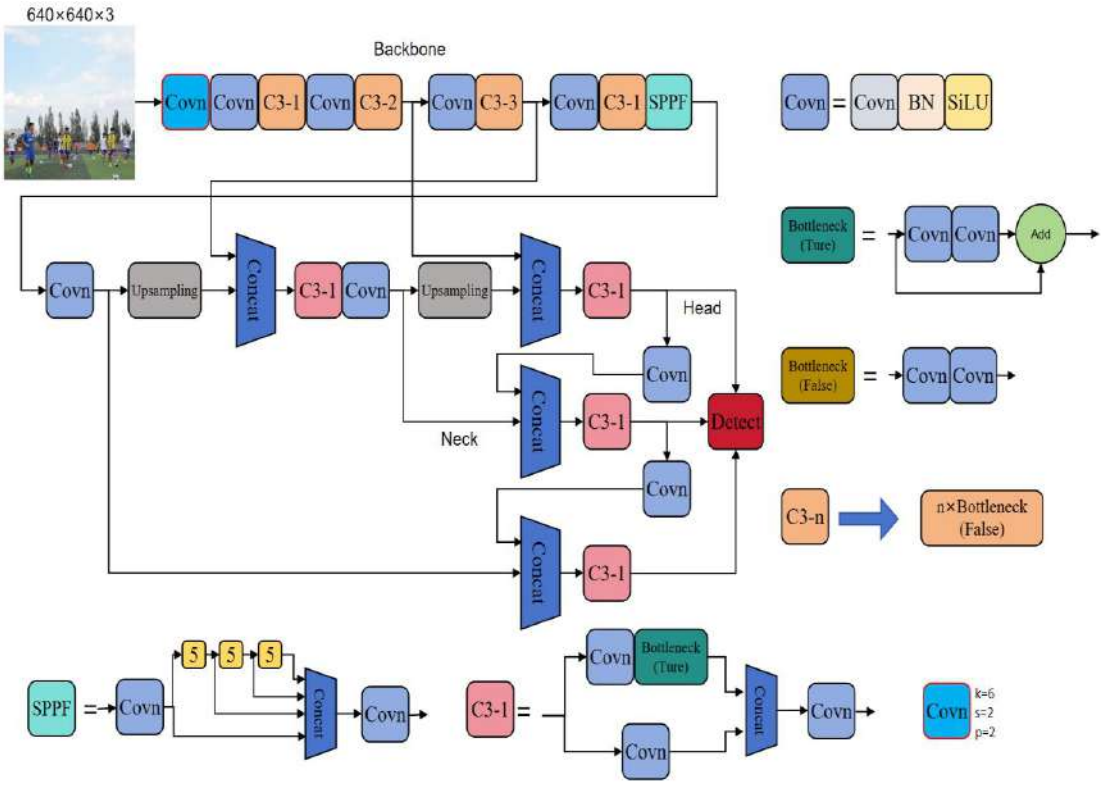

Figure 1: YOLOv5 Network Structure Diagram

The notation C3-n represents the C3 module, where n denotes the number of bottlenecks within the C3 module. When the bottleneck parameter is set to False, the C3 module consists of a cascade of a 1 × 1 Conv module and a 3 × 3 Conv module. When the bottleneck parameter is True, the C3 module employs a residual connection, adding the original input to the output after passing through two Conv modules. The configuration of the C3 module depends on the bottleneck parameter. If the C3 parameter is False, a bottleneck with the False parameter is used, and vice versa (indicated by the same color scheme in the figure). SPPF (Spatial Pyramid Pooling-Fast) is an upgraded version of SPP (Spatial Pyramid Pooling). While the outputs of SPPF and SPP are mathematically equivalent, SPPF reduces computational complexity and increases speed by cascading three 5 × 5 pooling layers.

2.2 The Improved YOLOv5 Algorithm

The YOLOv5 network chooses CSPDarknet53 as its backbone network, which achieves high detection accuracy but comes with a large number of parameters and substantial computational complexity. This makes it unsuitable for real-time target detection in football training scenarios. Therefore, to reduce the parameter count of the neural network and improve detection speed in football training scenarios, ensuring real-time and smooth detection, we simplified the YOLOv5 backbone network to the lightweight MobileNet while keeping the rest of the YOLOv5 network unchanged. The new network structure is shown in Figure 2.

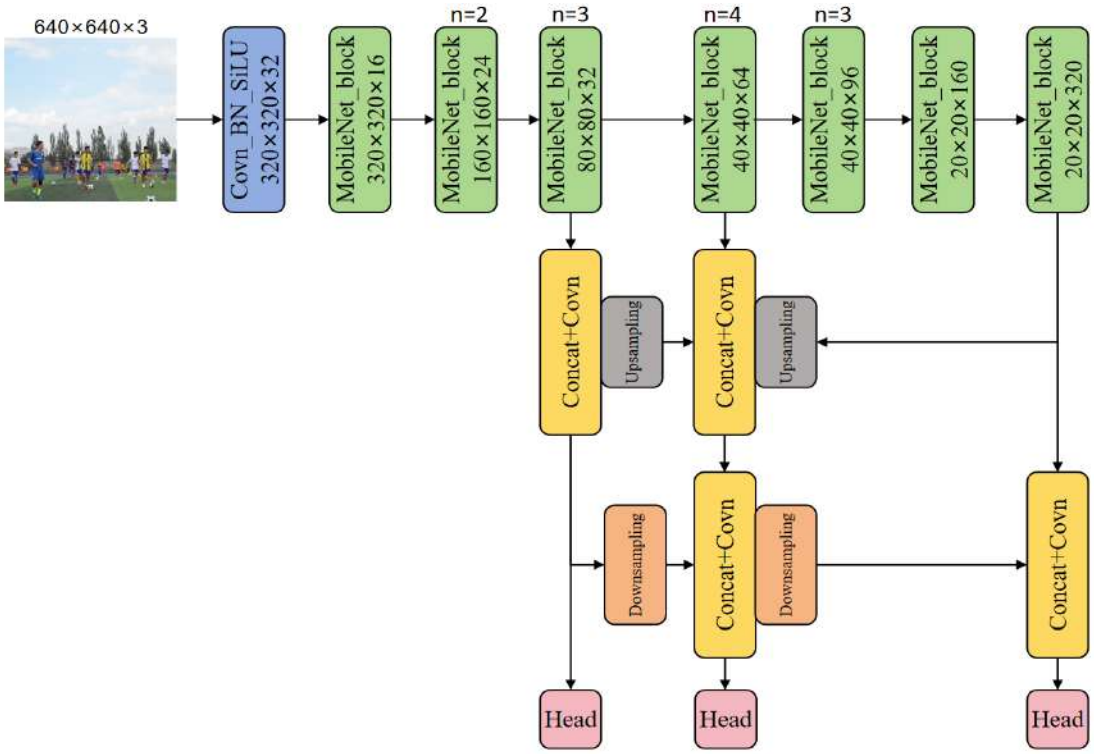

Figure 2: YOLOv5-MobileNet Network Structure

Mobile Net is a lightweight network structure characterized by using depth wise separable convolutions instead of standard convolutional kernels. It is suitable for scenarios with limited computing power, such as mobile devices, where CPU computation and fast, lightweight deployment are required. This approach maintains detection accuracy while accelerating inference speed. When conventional convolution operations extract image features, they typically need to simultaneously handle the spatial and channel features of the input image. Depending on the input features, the number of convolution kernels used may need to be adjusted, which can easily lead to a substantial increase in the number of model parameters. Depth wise separable convolutions consist of two parts: depth wise convolutions and pointwise convolutions. A conventional convolution kernel that accepts an input feature map of size $D_x \times D_y \times M$, where D_F is the spatial dimension and M is the number of input channels, produces an output feature map of size $D_x \times D_y \times N$, where N is the number of output channels. In a standard convolution operation, assuming the size of the convolution kernel is $D_K \times D_K$, the computational cost of a single convolution is given by:

$$compute_{cost} = D_K \cdot D_K \cdot M \cdot N \cdot D_x \cdot D_y \qquad (2)$$

Depthwise separable convolutions significantly reduce this computational cost by splitting the conventional convolution into a depthwise convolution followed by a pointwise convolution.

Depth wise separable convolutions first apply a convolutional kernel of size $D_K \times D_K$ to each channel of the input feature map individually. This process is known as depth wise convolution. Then, a 1 × 1 convolutional kernel is used to perform pointwise convolution, linearly combining the convolved feature maps. As illustrated in Figure 3:

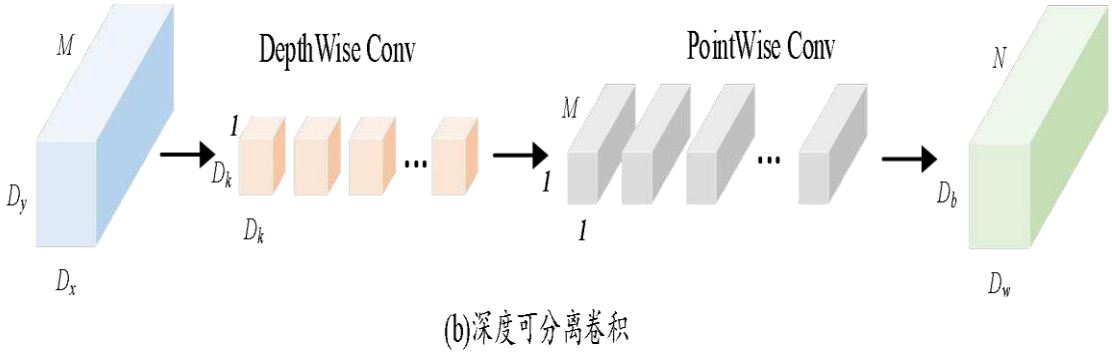

Figure 3: Deep separable convolution

The computational effort required for deeply separable convolution is:

$$compute_{cost}^{dsc} = D_K \cdot D_K \cdot M \cdot D_x \cdot D_y + M \cdot N \cdot D_x \cdot D_y \qquad (3)$$

Dividing Equation 3 by Equation 2 yields.

$$\frac{compute_{cost}^{dsc}}{compute_{cost}} = \frac{1}{N} + \frac{1}{K^2} \qquad (4)$$

When the number of convolution kernels becomes sufficiently large, the ratio's first part, $\frac{1}{N}$, can be neglected. Therefore, the speed of depth wise separable convolutions is approximately proportional to the square of the kernel size. For instance, when the kernel size is 3 × 3, the speed of depth wise separable convolutions is approximately nine times faster than that of standard convolutions. Consequently, and as demonstrated by subsequent experimental results, the detection speed of the improved YOLOv5 is significantly increased.

2.3 Neck Structure Improvement

According to the COCO dataset standards, the definition of small objects can be analyzed from two perspectives: relative size and absolute size. From the absolute size perspective, objects smaller than 32 × 32 pixels can be considered small objects. From the relative size perspective, objects whose length and width are each less than 10% of the original image size can also be considered small objects. As shown in Figure 1, the output feature maps of YOLOv5 are divided into three categories: a 20 × 20 × 1024 feature map for detecting large objects, a 40 × 40 × 512 feature map for detecting medium

objects, and an 80 × 80 × 256 feature map for detecting small objects. After a series of down-sampling operations, the resolution of the output layer's feature maps is much lower than that of the original input image, which can easily result in missing small objects (G. Jin, 2022). In neural networks, the neck structure lies between the backbone network and the prediction head, with the purpose of aggregating as much information extracted by the backbone network as possible before making predictions. This structure is particularly useful for transmitting small object information, as it can prevent the loss of small object information when it flows to higher layers. The specific approach is to up sample the resolution of the feature maps again and aggregate feature maps of different sizes from the backbone network, enabling better detection performance. This study refers to the weighted Bidirectional Feature Pyramid Network (BiFPN) (Tan, Pang, & Le, 2020) structure to improve the YOLOv5 neck. The neck part of YOLOv5 uses FPN and PAN. The structures of FPN, PAN, and BiFPN are shown in Figure 4, respectively. FPN and PAN treat all feature maps as homogeneous, making the roles of feature maps of different resolutions the same in the feature pyramid. However, in practice, the impact of different-sized input features on the output features is often different. This study assigns a learnable weight parameter to each input feature, allowing the network to automatically adjust the influence of each input feature on the output results. The weight settings are shown in Equations 5 and 6.

$$P_6^{td} = Conv\left(\frac{w_1 \cdot P_6^{in} + w_2 \cdot Resize\left(P_7^{in}\right)}{w_1 + w_2 + \varepsilon}\right) \quad (5)$$

$$P_6^{out} = Conv\left(\frac{w_1' \cdot P_6^{in} + w_2' \cdot P_6^{td} + w_3' \cdot Resize\left(P_5^{out}\right)}{w_1' + w_2' + w_3' + \varepsilon}\right) \quad (6)$$

P_6^{td} represents the intermediate features from the 6-th layer in the top-down pathway. P_6^{out} represents the output features from the 6-th layer in the bottom-up pathway. Other features follow a similar principle.

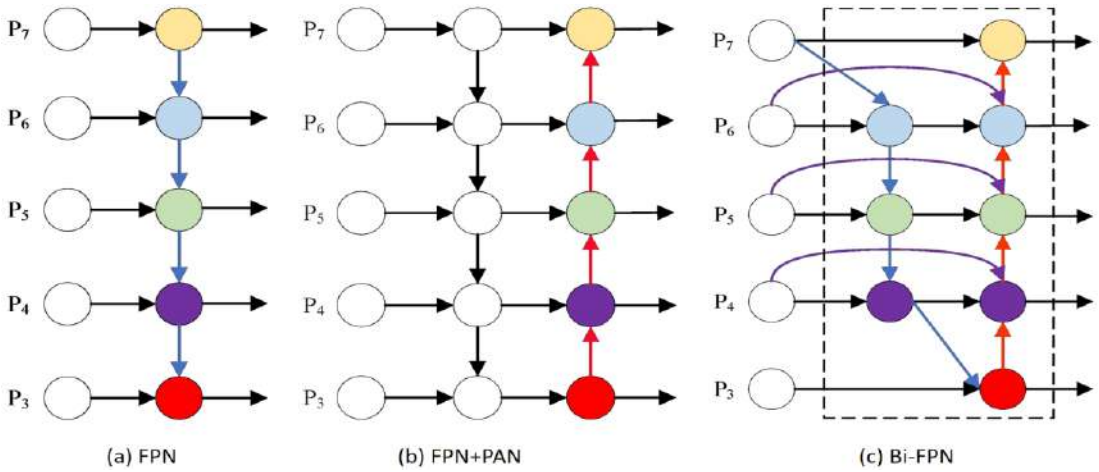

Figure 4: Schematic diagram of Bi-FPN structure

To reduce the required computational load, the improved neck replaces standard convolutions with depthwise separable convolutions. Additionally, for nodes with only one input edge, since these nodes do not perform feature fusion and do not impact the feature network, they can be removed without compromising the feature fusion effect, thereby simplifying the network. Subsequently, based on specific hierarchical conditions, an edge is added from the original input to the output nodes to integrate more features. Finally, each bidirectional feature path is considered as a layer of the feature network (Sun, Zhao, Zhao, Jia, & Cao, 2022).

2.4 Integrating Attention Mechanism

In football training scenarios, complex background interferences such as billboards and spectators can negatively impact the detector's effectiveness. To address this, we introduce an attention mechanism into the YOLOv5 network structure. The benefit of this approach is that it enhances the network's focus on the target, ignoring irrelevant background factors, thereby improving the accuracy of target detection (Zhang et al., 2020).

SENet, short for Squeeze-and-Excitation Network, is a channel attention mechanism that focuses on the correlations between the channel dimensions of an image, examining the impact of each channel on the network. Hu et al. achieved the highest detection accuracy on ImageNet2017 using convolutional neural networks with the SENet structure (Jing & Xiaoqiong, 2021).

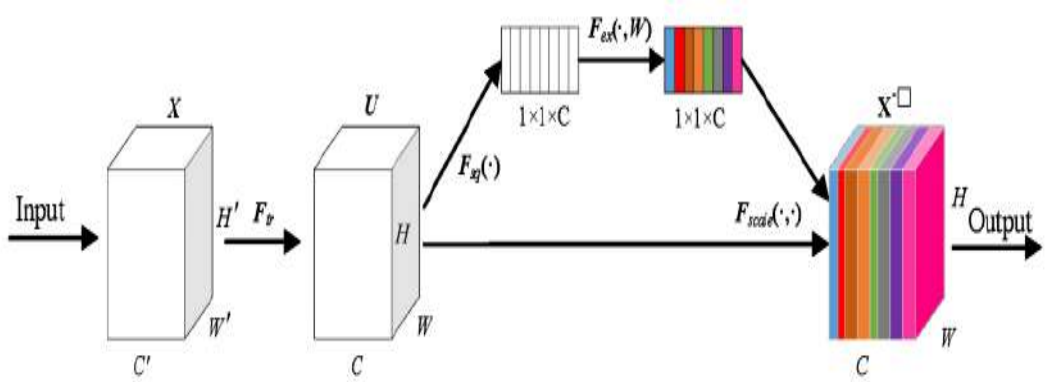

Figure 5: SE Attention Mechanism

Convolutional neural networks use a large number of convolutional kernels to extract spatial and channel information within local receptive fields and fuse them to construct target feature information. The size of convolutional kernels is typically 3 × 3 or 5 × 5. As convolutional layers stack, the receptive field of deep networks expands, leading to many multi-level dependencies. This results in insufficient feature extraction at the image edges, and positional information at different distances is centrally transmitted through the network

layers, causing long-distance dependency issues (Xing, Ai, Liu, & Lao, 2010). SENet proposes a mechanism to re-calibrate network features, learning to selectively emphasize informative features using global information and suppress less useful ones, thereby achieving a reconstruction of the convolutional feature channel dimensions.

As shown in Figure 5, SENet primarily includes four steps: F_{tr}, F_{sq}, F_{ex}, and F_{scale}. F_{tr} is a transformation step aimed at modeling the interdependencies between channels before the Squeeze and Excitation recalibration filters the output. F_{tr} can be simply viewed as a convolution operation, and the transformation formula is shown as follows:

$$F_{tr}: X \to U, X \in R^{H' \times W' \times C'}, U \in R^{H \times W \times C} \quad (7)$$

F_{sq} is the Squeeze operation, which aims to compress the global spatial information into a channel descriptor. Essentially, this operation uses global average pooling to aggregate the information across channels. Typically, it compresses U in the $H \times W$ spatial dimensions to obtain the statistical information z. The calculation formula for z is as follows:

$$z_c = F_{ex}(u_c) = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} u_c(i, j) \quad (8)$$

F_{ex} is the excitation operation, also known as adaptive recalibration. After obtaining the compressed information through the Squeeze operation, the excitation operation re-calibrates the channels by extracting the dependencies between them, satisfying the nonlinearity and non-mutual exclusion relationships between features in different channels. The formula is as follows:

$$s = F_{ex}(z, W) = \sigma\big(g(z, W)\big) = \sigma\big(W_2 \delta(W_1 z)\big) \quad (9)$$

Where, δ represents the ReLU activation function, and σ represents the sigmoid activation function. F_{scale} uses the activation function to re-adjust the transformed output U. Each channel of U is multiplied by the corresponding weight obtained from the SENet module to produce the output $\tilde{x}_c$. The calculation formula is as follows:

$$\tilde{x}_c = F_{scale}(u_c, s_c) = s_c \cdot u_c \quad (10)$$

There are two ways to add the attention module: one is to add it at the end of the original networkk structure's backbone, before the SPPF; the other is to replace all the C3 modules in the backbone with attention mechanism modules, as shown in Figure 6.

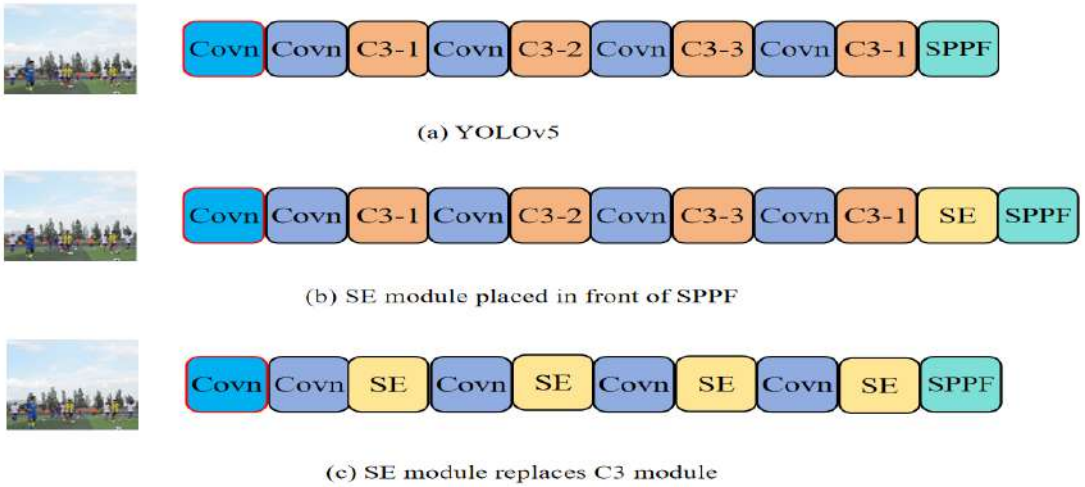

Figure 6: Schematic diagram of the two SE attention module embedding methods

ECA (Efficient Channel Attention) is an improvement over the SE attention mechanism. The SE attention mechanism has many parameters, and convolutional neural network feature maps are typically considered as three-dimensional tensors, where the first dimension is the feature channels, and the second and third dimensions are the spatial width and height of the feature map. The core idea of the ECA attention mechanism is to generate an attention coefficient for each feature channel to focus on channels with important information. Its illustration is shown in Figure 7. First, the input feature map undergoes a global average pooling operation. Then, the pooled features are processed with a one-dimensional convolution to generate a weight for each feature point. Finally, these weights are combined with the original feature map through a residual connection to obtain the output attention feature layer.

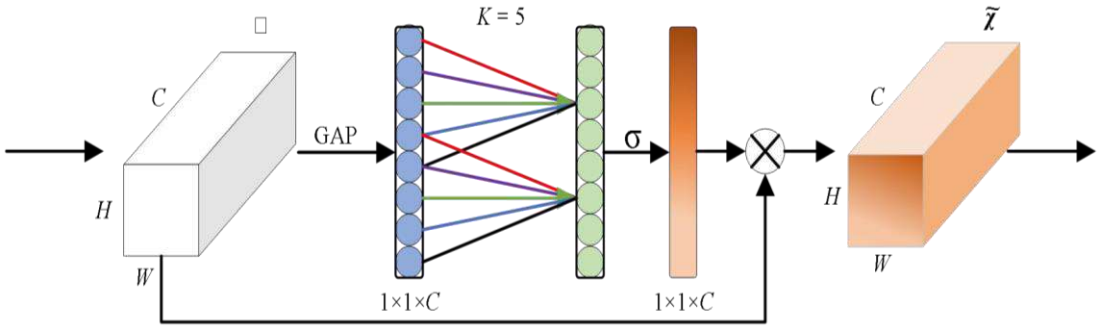

Figure 7: ECA Attention Module

The main contributions of the ECA attention mechanism are as follows:

1. Using 1D Convolution to Capture Inter-Channel Relationships: This allows the features of different channels to influence each other, thereby enhancing important channels and suppressing less important ones, while avoiding dimensionality reduction and reducing information loss.

1. Proposing a Local Cross-Channel Information Interaction: This includes

several steps:

◆ First, for each feature channel, an adaptive linear transformation generates a corresponding feature descriptor, indicating the importance of that channel.

◆ Then, for each feature map, all channel descriptors are averaged to generate a global descriptor.

◆ Next, a single-layer fully connected network maps the global descriptor into a vector of the same dimension as the number of channels, representing the global features.

◆ Finally, for each feature channel, the global feature vector is element-wise multiplied with the channel descriptor to obtain the final feature representation.

The advantages of the ECA attention mechanism include significantly improving the quality and expressive power of the feature map while reducing redundant information and noise. Additionally, since the ECA attention mechanism only uses global descriptors and channel descriptors for simple weighted summation and replaces the fully connected operation in SE attention with 1D convolution, it has relatively high computational efficiency. This makes it suitable for large-scale image datasets and deep network structures. The two embedding methods of the ECA attention mechanism are illustrated in Figure 8.

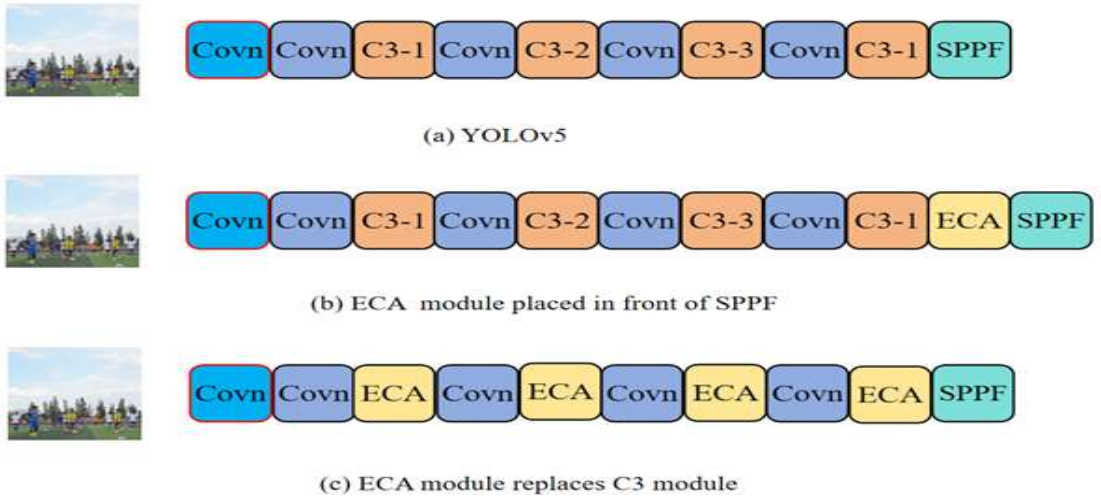

Figure 8: Schematic diagram of the two ECA attention module embedding methods

CA (Coordinate Attention) is an attention mechanism commonly used in mobile applications. It is a type of channel attention mechanism that integrates positional information into channel attention with almost no additional computational overhead. Unlike channel attention that uses 2D global pooling to convert feature tensors into 1D feature vectors, CA attention separates

channel attention into two 1D feature encoding stages, where information from two spatial directions is fused during the encoding stage. The schematic diagram of the CA attention module is shown in Figure 9.

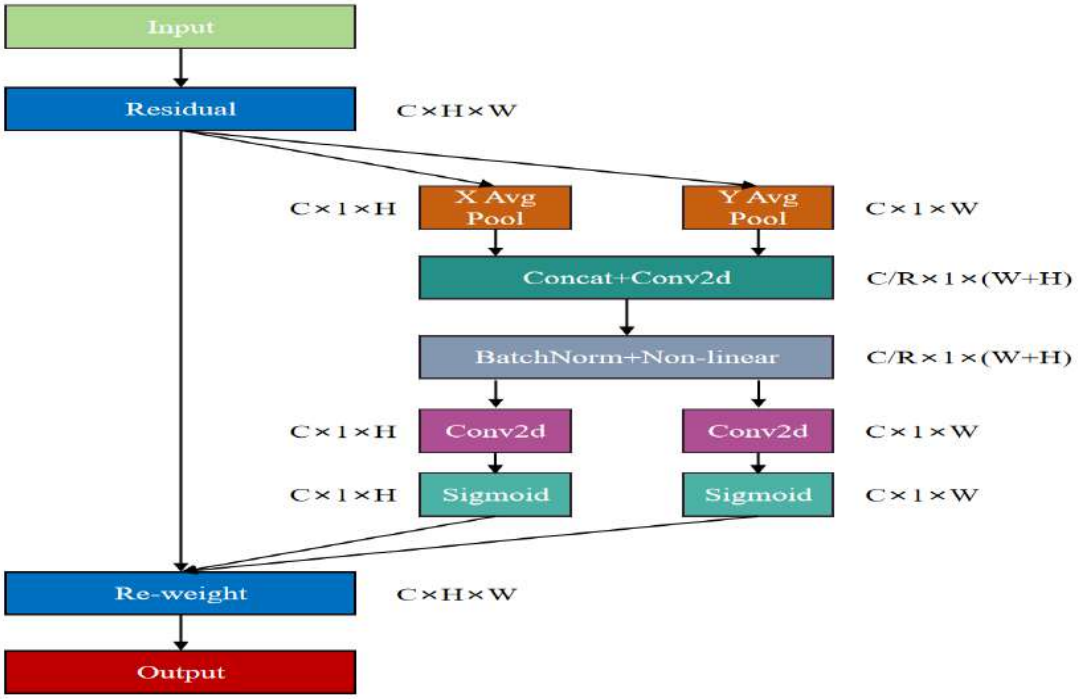

Figure 9: CA Attention Module

The input to the CA attention module is a feature map F of size $C \times H \times W$. After entering the module, F first passes through a residual block, maintaining its size as $C \times H \times W$, and then splits into three branches. The first branch uses $F1$as the spatial feature to retain, while the second and third branches undergo attention weight transformations. Meanwhile, the second and third branches also pass through an average pooling layer, resulting in $F2\_1$ and $F3\_1$.

The purpose of this step is to separate the image features into two dimensions: H and W. The sizes of $F2\_1$ and $F3\_1$ are $C \times H \times 1$ and $C \times W \times 1$, respectively. Next, $F2\_1$ and $F3\_1$ are concatenated along the width and height dimensions, respectively, and then undergo a 2D convolution operation. In this step, the two branches are temporarily combined. The combined feature size is $\left({}^{C}/_{r}\right) \times 1 \times (W + H)$). Then, the combined features are batch normalized, activated using a nonlinear activation function, and passed through another 2D convolution and activation, resulting in two attention maps of size $C \times 1 \times W$ and $C \times 1 \times H$. These attention maps are used to weight the spatial features $F1$, meaning the weighted attention maps are multiplied with the original feature map $F1$ to obtain the final output $F1 \times F2 \times F3$, resulting in a feature map of size $C \times H \times W$. The two embedding methods

of the CA attention mechanism are illustrated in Figure 10.

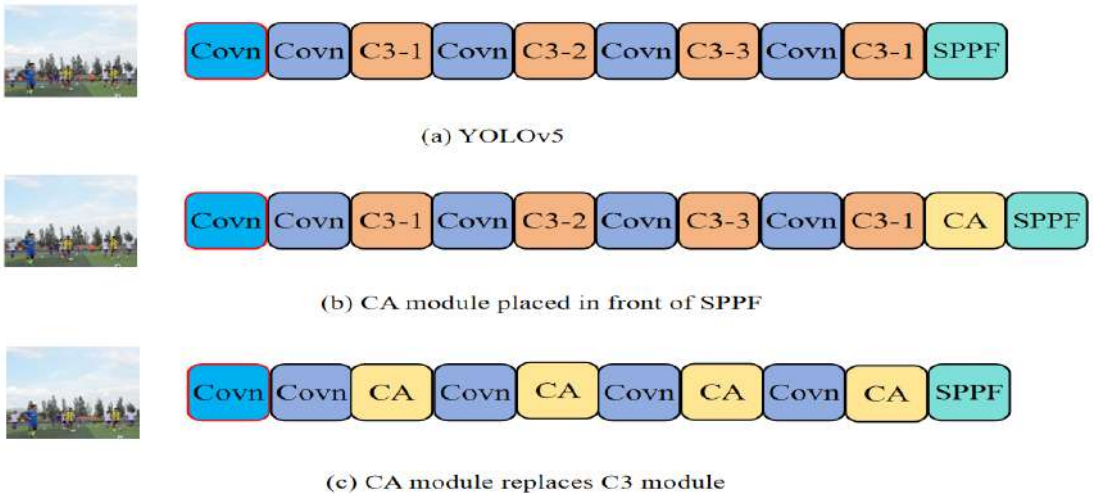

Figure 10: Schematic diagram of the two CA attention module embedding methods

3. Experiments

3.1 Experimental Data

The dataset used in the experiment is the Soccer DB dataset, collected and annotated by Cui et al. (Jiang, Cui, Chen, Wang, & Xu, 2020). The images in the dataset are sourced from 346 football matches, totaling 668.6 hours of football match footage. Among these, 270 matches were selected from the 2014 to 2017 seasons of the six major European leagues, 76 matches from the 2017 to 2018 season of the Chinese Super League, and parts of the recent three World Cups, providing a broad representation.

To increase the difficulty of the dataset, the authors first crawled 24,475 images covering various football match scenes from the internet. These images were used to initially train a detector. The detector was then employed to perform object detection on the match footage, annotating the images with the target bounding box information. The frames with the poorest detection performance were selected as the final target detection dataset. The final dataset for object detection contains 45,732 images, each with a size of 1280x720 pixels, comprising approximately 700,000 target bounding boxes divided into three categories: football, player, and goal. The specific distribution is shown in Table 1.

Table 1: Distribution of labeled boxes in the data set

CATEGORY	QUANTITIES
PLAYERS	643581
SOCCER	45160
GOALS	13355
TOTAL	702096

The dataset was divided into training and test sets in an 8:2 ratios. The dataset was annotated using the COCO dataset format, with a text file generated for each image containing the target information present in that image. Each line in the text file stores the information of one target bounding box, divided into several columns. The first column indicates the category of the target, while the second to fifth columns represent the center coordinates and the width and height of the bounding box, respectively.

3.2 Experimental Setup and Parameters

The experimental environment used in this study includes the following: the operating system is Ubuntu 16.04, the processor is an Intel® Core™ i7-9750H CPU @ 2.60GHz, and the graphics card is an NVIDIA GeForce RTX 2080 Ti with 11GB of memory. The deep learning framework employed is PyTorch.

3.3 Evaluation Metrics

For the detection results of the model, based on the combination of the actual target category and the predicted category, the results can be divided into True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN). TP represents a case where the IoU (Intersection over Union) of the bounding box is greater than 0.5. FP represents cases where the IoU is less than 0.5 or the bounding box is duplicated.

FN indicates that the bounding box failed to detect the target or the IoU is greater than 0.5 but the detection category is incorrect. TN represents cases where no bounding box is detected in locations without a target, which does not affect the detection performance of the target, and hence is not used. The confusion matrix for the classification of detection results is shown in Table 2.

Table 2: Confusion Matrix of Detection Results

THE REAL SITUATION	**PROJECTED RESULTS**	
	positive example	negative example
POSITIVE EXAMPLE	True Positive(TP)	False Positive(FP)
COUNTER-EXAMPLE	True Negative(TN)	False Negative(FN)

Checking accuracy (precision) is defined as:

$$Precision = \frac{TP}{TP+FP} \tag{11}$$

The search rate (recall) is defined as:

$$Recall = \frac{TP}{TP+FN} \tag{12}$$

The commonly used standard for evaluating the effectiveness of object detection is Average Precision (AP). AP was initially proposed by the PASCOL VOC object detection competition. It refers to the area under the Precision-Recall (PR) curve, with precision on the horizontal axis and recall on the vertical axis. The specific algorithm for calculating AP is shown in the following formula:

$$P = \frac{1}{11}\sum_{r\in\{0,0.1,\cdots,1\}} p_{interp}(r) \quad (13)$$

Where，

$$p_{interp}(r) = \max_{\tilde{r}:\tilde{r}\geq r} p(\tilde{r}) \quad (14)$$

It represents the precision for each recall value, taking the maximum precision value to the right of that point as the precision value for that point. An image often contains multiple categories of targets. The mean Average Precision (mAP) is obtained by summing the average precision of all target categories and then taking the mean value. The formula for calculating mAP is as follows:

$$mAP = \frac{\sum_{c=1}^{C} AveP(c)}{C} \quad (15)$$

Where C represents the number of categories, $c$represents a specific category, and $AveP(c)$ represents the average precision for category c.

3.4 Experimental Results and Analysis

(1) The comparison of model complexity and performance between YOLOv5-MobileNet and the original YOLOv5 after training on the dataset is shown in Table 3:

Table 3: Comparison of YOLOv5 and YOLOv5-MobileNet Performance

MODEL	MAP@.5	FPS (GPU)	FPS (CPU)	GFLOPS	QUANTITY OF PARTICIPANTS
YOLOV5	0.913	67.71	4.71	16.4	7.08M
YOLOV5-MOBILENET	0.887	79.08	7.28	6.3	3.57M

As shown in the table above, YOLOv5-MobileNet achieved a 2.6 percentage point reduction in mAP compared to the original YOLOv5, indicating that the simplified backbone network's ability to extract target features is somewhat diminished, leading to a decrease in accuracy. In terms of model parameters and complexity, YOLOv5-MobileNet has approximately half the number of parameters, and the computational load is about 38% of the original,

demonstrating that the simplification of the backbone network effectively reduces the model's complexity. Furthermore, the detection speed of YOLOv5-MobileNet is 17% higher than that of the original model on a GPU, and the detection speed on a CPU is increased by half. This suggests that although the simplified backbone network results in a slight decrease in detection accuracy, it significantly reduces the number of parameters and the model size, enhancing detection speed. This makes the model more suitable for the target detection requirements of football training scenarios.

Table 4: Performance Comparison Before and After Improving Small Object Detection

MODEL	MAP@.5
YOLOV5	85.8
YOLOV5-MOBILENET	87.5

As shown in Table 4, after optimizing the neck part of YOLOv5, the model's detection accuracy for small objects increased by nearly two percentage points due to the fusion of multi-scale features.

(3) Comparison of model performance with the integration of the attention mechanism.

Table 5: (a) Comparison of Attention Models

SIZE	C3	SE	ECA	CA	MAP@.5
YOLOV5S	✓				0.870
		✓			0.873
			✓		0.890
				✓	0.887
	✓	✓			0.890
	✓		✓		0.887
	✓			✓	0.888
YOLOV5M	✓				0.918
		✓			0.916
			✓		0.912
				✓	0.931
	✓	✓			0.916
	✓		✓		0.912
	✓			✓	0.917
YOLOV5L	✓				0.942
		✓			0.944
			✓		0.962
				✓	0.928
	✓	✓			0.937
	✓		✓		0.941
	✓			✓	0.941

The networks with embedded attention mechanisms were trained on the dataset, and the model performance was recorded, as shown in Table 5. Here, C3, SE, ECA, and CA represent different modules of the network. Checking only the C3 row indicates the basic, unmodified YOLOv5 structure. Checking only the attention mechanism module row indicates that the C3 modules were replaced with the selected attention mechanism modules. Checking both the attention mechanism module and the C3 module rows indicates that the attention mechanism module was added before the SPPF module without replacing the C3 modules. As shown in the table, as the neural network model size increases, the detection accuracy also improves. For the same model size, embedding attention mechanisms improves detection accuracy, with networks embedding attention mechanisms achieving a two to three percentage point increase in accuracy compared to the original YOLOv5 network. It is also noted that embedding attention mechanisms helps to offset the accuracy gap caused by model size differences. Among the various attention mechanisms, the ECA attention mechanism shows a more significant improvement in accuracy overall. Additionally, from the perspective of the number of parameters, if the detection accuracy is similar, replacing the C3 modules with attention mechanism modules is preferred because it reduces the number of parameters.

4. DeepSORT-Based Tracking Model

In our tracking system, we use DeepSORT as the baseline tracking model and improve it by incorporating the Unscented Kalman Filter. The outputs from the improved YOLOv5 detection model, which include bounding boxes, are used as the input for the tracking system. The final output is a video stream where each target in every frame is assigned a tracking box. For the input video stream, the improved YOLOv5 detector first detects the targets and predicts their locations, generating bounding boxes. These detected bounding boxes are then converted into detections recognized by DeepSORT. Based on the detection results, tracking is performed using DeepSORT. The multi-object tracking algorithm accepts the target bounding box information detected by the object detector, uses the Kalman Filter to form motion trajectories, and then employs the Hungarian Algorithm to match the predicted bounding boxes with the detected bounding boxes in the current frame. Successful matching indicates successful tracking, and the Kalman Filter is used to update the state. The baseline tracking steps are as follows: 1. The YOLOv5 detector detects each frame of the video stream sequentially, outputting the labels and confidence scores of the players in the current frame. Simultaneously, the current frame is used to initialize the Kalman Filter to predict the target's position in the next frame. 2. In the next frame, the YOLOv5 detector obtains the label box information, and the Hungarian Algorithm generates a cost matrix.

1. If the predicted player label box from the previous frame appears in the current frame, it is found in the cost matrix, and its position information is used

as the basis for predicting the target's location in the next frame using the Kalman Filter. 3. When a new player appears in a frame, the detection box of the new player is referenced by the cost matrix, and the Kalman Filter generates a new tracking object to track the newly appeared player. 4. When a player is tracked in consecutive frames, the tracking is considered successful. If a successfully tracked player fails to match the corresponding detection frame's bounding box, the tracking is deemed a failure or out of the video detection area and is no longer tracked. 5. Repeat the above steps until the video stream ends. In the DeepSORT tracking model, it is important to note that the input bounding boxes are detected by the YOLOv5 detector for each frame of the video stream. The Unscented Kalman Filter acts as a predictor, handling the target state described by a series of observations over time through the target's state vector. The tracking system includes analyzing detected targets, assigning a unique ID to each target, following their motion paths, and maintaining consistent IDs. Additionally, in real-time target tracking, the displacement of detected targets between adjacent frames is typically considered ideal. It can be represented by a uniform motion model unaffected by other object movements. DeepSORT is a detection-based multi-object tracking algorithm with excellent performance in multi-object tracking, making it suitable for real-time multi-player statistics tasks.

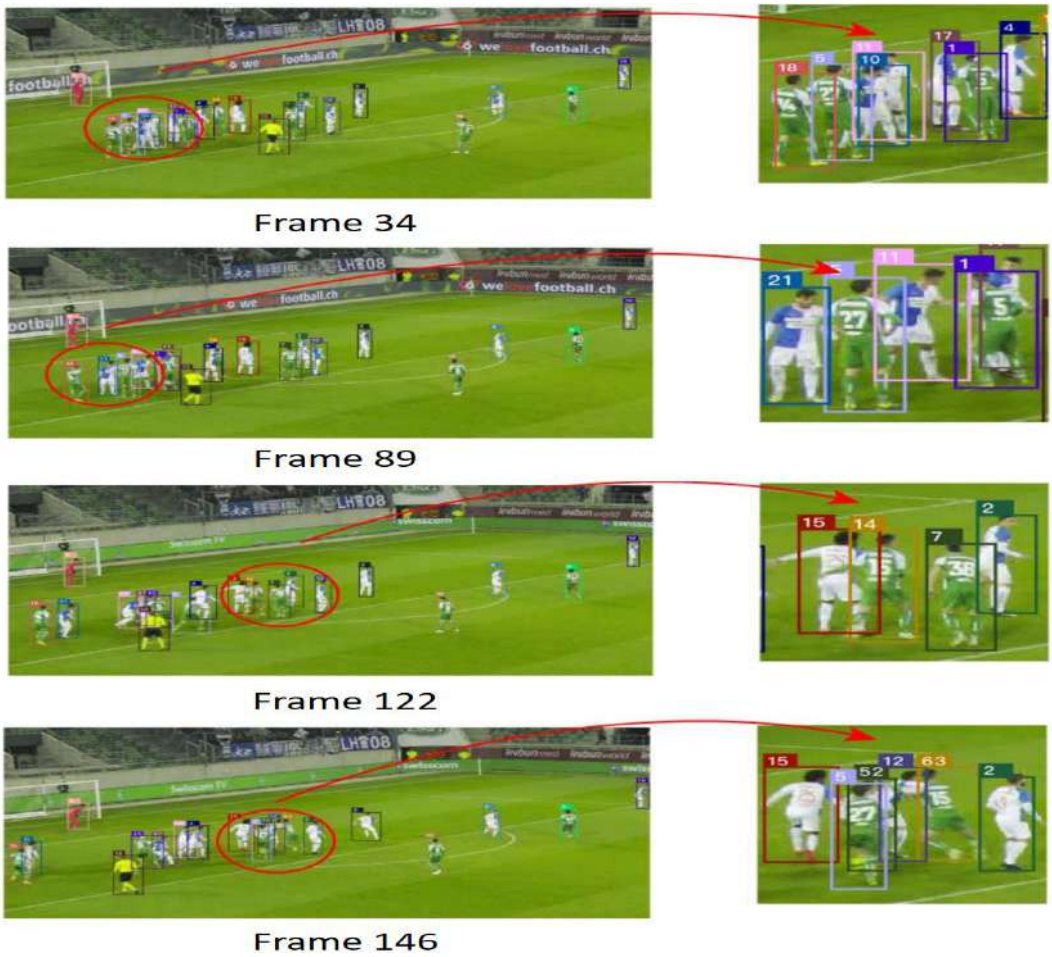

Figure 11: Tracking results of the DeepSORT algorithm

As shown in Figure 11, the original tracker's performance is displayed on frames 34, 89, 122, and 146 of a particular video segment in the dataset. The figure illustrates that in scenarios with densely distributed players, the target with tracking ID 10 running from right to left in frame 34 changes its ID in frame 89. Additionally, the targets with IDs 2 and 7 in frame 122 also experience an ID switch in frame 146.

5. Application of the Football Basic Training System

Compared to football matches, university football training scenarios are more challenging due to limitations in the venue and equipment, as well as more complex conditions regarding lighting, player attire, and on-field activities. A good training system should be adaptable to both tactical training and practice matches. To verify the adaptability of the university football basic training system constructed in this study, various training scenarios were selected to test the system's target detection capabilities.

5.1 Daytime Lighting Environment

In university football basic training, the daytime lighting environment is prone to challenges such as large shadows caused by weather and buildings, which can impact target detection. Additionally, since athletes do not wear uniform attire during regular training, this results in highly dispersed athlete characteristics, further increasing the difficulty of target detection.

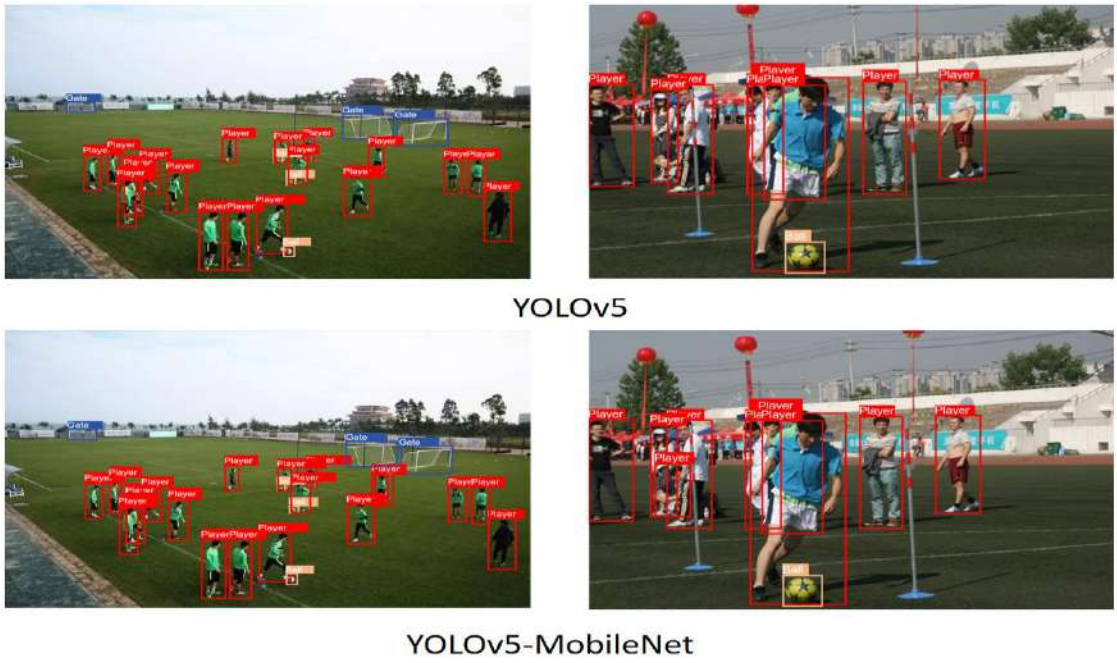

Figure 12: Comparison of the performance of the original YOLOv5 model and the YOLOv5-MobileNet model under daytime lighting conditions

As shown in Figure 12, the original YOLOv5 model can effectively

recognize most targets in a daytime lighting environment. However, it may miss detections when athletes are occluded. In contrast, the YOLOv5-MobileNet model can detect athletes even when they are occluded or in a sitting position, demonstrating the model's superior performance. The improved model is better suited for handling occlusions that occur during line-up training in basic football training scenarios.

5.2 Nighttime Lighting Environment

Nighttime training environments are very common in university football basic training. Due to equipment limitations, the lighting conditions for nighttime training cannot be compared to those of competitive matches. Additionally, the varying placement of lighting equipment introduces extra light source interference in nighttime football training videos. This is especially problematic when light sources directly illuminate the video capture equipment, severely affecting video quality and significantly increasing the difficulty of target detection.

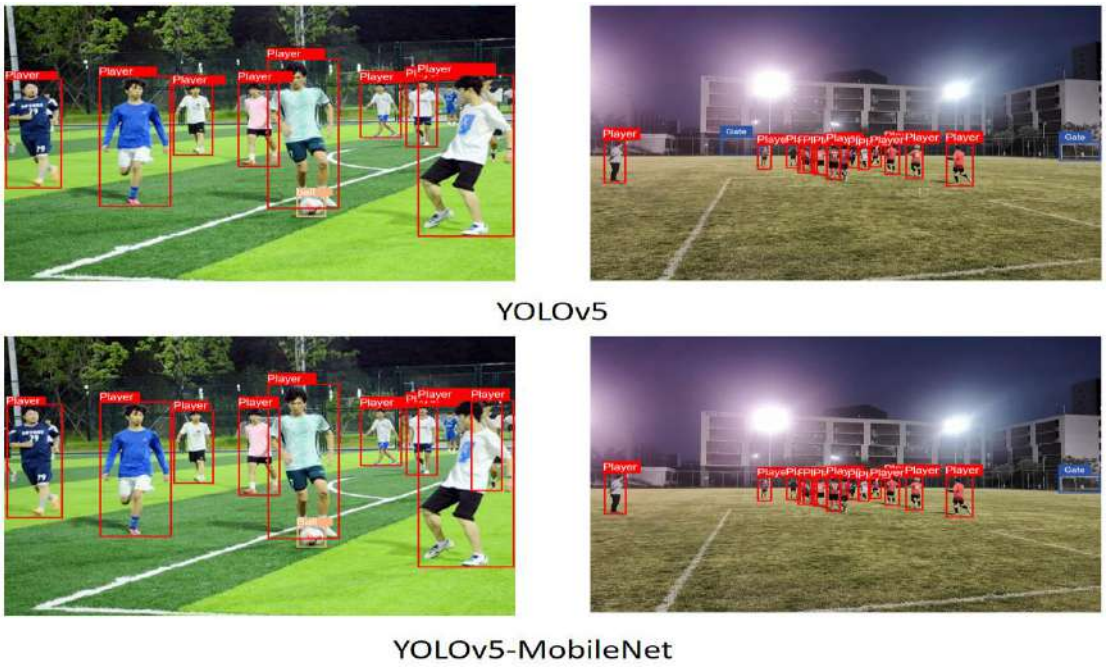

Figure 13: Comparison of the performance of the original YOLOv5 model and the YOLOv5-MobileNet model under nighttime illumination conditions

As shown in Figure 13, both YOLOv5 and YOLOv5-MobileNet can effectively detect athletes in well-lit environments, as displayed in the left image. However, when athletes are occluded, especially with large occlusions, YOLOv5 tends to miss detections, whereas the YOLOv5-MobileNet model demonstrates better detection accuracy. In the left image of Figure 13, under nighttime lighting conditions with strong light interference, YOLOv5 incorrectly detects the field fence near the strong light source as a goal. These experiments verify the effectiveness of the proposed model and its superior adaptability to

various environments.

6. Conclusion

To address the issue of decreased detection accuracy in traditional object detection caused by occlusion, uneven lighting, and small object size, this paper simplifies the backbone structure of the YOLOv5 network. The original CSPDarkNet53 structure was streamlined into the Mobile Net structure with depth wise separable convolutions, reducing the number of parameters and effectively improving the detection speed. To tackle the problems of target occlusion and uneven lighting conditions, different attention mechanisms were embedded into the network model, enhancing the model's target detection capabilities. For the problem of small object size, a multi-scale output layer was reconstructed to integrate multi-scale features, effectively reducing the issues of missed and incorrect detections of small targets. The improved YOLOv5 model was tested on public datasets, and experimental results indicate that the proposed model exhibits superior detection performance. The Deep SORT tracking algorithm improves the position prediction capability for non-linear moving targets and enhances the algorithm's target matching ability. Testing the improved Deep SORT algorithm on the dataset demonstrated good tracking performance. In this study, the proposed improved YOLOv5 algorithm was trained from scratch using randomly initialized parameters, resulting in a longer training time. Future work could employ transfer learning to both improve detection accuracy and reduce training time. Additionally, the dataset used for target detection did not annotate target sizes, suggesting that the detection performance for small targets could be further improved. While the multi-object tracking algorithm achieved good tracking results, there is still room for improvement in scenarios with densely distributed and highly interlaced players.

Acknowledgement

General project of Zhejiang philosophy and social science planning project: Research on brand culture and construction path of China Super League Football League (Grant No.20NDJC184YB).

Reference

Adham, A. A., Othman, A. M., Karim, S., George, A., Natalie, S., Rabih, A. C., & Efthymios, D. A. (2022). Acute Deep Vein Thrombosis Involving the Inferior Vena Cava: Interventional Perspectives. *Vascular & Endovascular Review, 5*.

Diwan, K., Bandi, R., Dicholkar, S., & Khadse, M. (2023). *Football Player and Ball Tracking System Using Deep Learning.* Paper presented at the Proceedings of International Conference on Data Science and Applications: ICDSA 2022, Volume 1.

Dwijayanto, M. R., Kurniawan, S., & Sugandi, B. (2019). *Real-Time Object*

Recognition for Football Field Landmark Detection Based on Deep Neural Networks. Paper presented at the 2019 2nd International Conference on Applied Engineering (ICAE).

Guntuboina, C., Porwal, A., Jain, P., & Shingrakhia, H. (2021). Deep learning based automated sports video summarization using YOLO. *ELCVIA Electronic Letters on Computer Vision and Image Analysis, 20*(1), 99-116.

Huang, Z., Zhang, P., Liu, R., & Li, D. (2023). An Improved YOLOv3-Based Method for Immature Apple Detection. *IECE Transactions on Internet of Things, 1*(1), 9-14.

Jiang, Y., Cui, K., Chen, L., Wang, C., & Xu, C. (2020). *Soccerdb: A large-scale database for comprehensive video understanding.* Paper presented at the Proceedings of the 3rd International Workshop on Multimedia Content Analysis in Sports.

Jin, G. (2022). Player target tracking and detection in football game video using edge computing and deep learning. *The Journal of Supercomputing, 78*(7), 9475-9491.

Jin, X., Tong, A., Ge, X., Ma, H., Li, J., Fu, H., & Gao, L. (2024). YOLOv7-Bw: A Dense Small Object Efficient Detector Based on Remote Sensing Image. *IECE Transactions on Intelligent Systematics, 1*(1), 30-39.

Jing, H., & Xiaoqiong, X. (2021). Sports image detection based on FPGA hardware system and particle swarm algorithm. *Microprocessors and Microsystems, 80*, 103348.

Rangappa, S., Li, B., & Qian, R. (2021). *Tracking and identification for football video analysis using deep learning.* Paper presented at the Thirteenth International Conference on Machine Vision.

Sun, P., Zhao, X., Zhao, Y., Jia, N., & Cao, D. (2022). Intelligent optimization algorithm of 3d tracking technology in football player moving image analysis. *Wireless Communications and Mobile Computing, 2022*(1), 5509095.

Tan, M., Pang, R., & Le, Q. V. (2020). *Efficientdet: Scalable and efficient object detection.* Paper presented at the Proceedings of the IEEE/CVF conference on computer vision and pattern recognition.

Wang, J., Zhang, T., Cheng, Y., & Al-Nabhan, N. (2021). Deep Learning for Object Detection: A Survey. *Computer Systems Science & Engineering, 38*(2).

Xing, J., Ai, H., Liu, L., & Lao, S. (2010). Multiple player tracking in sports video: A dual-mode two-way bayesian inference approach with progressive observation modeling. *IEEE Transactions on image Processing, 20*(6), 1652-1667.

Yang, J., & Gapar, Y. (2024). Improved Object Detection Algorithm Based on Multi-scale and Variability Convolutional Neural Networks. *IECE Transactions on Emerging Topics in Artificial Intelligence, 1*(1), 31-43.

Yang, M., & Fan, X. (2024). YOLOv8-Lite: A Lightweight Object Detection Model

for Real-time Autonomous Driving Systems. *IECE Transactions on Emerging Topics in Artificial Intelligence, 1*(1), 1-16.

Yasaman, K., & Caitlin, W. H. (2023). Acute Complicated Type B Aortic Dissection: Do Alternative Strategies Versus Central Aortic Repair Make Sense? *Vascular & Endovascular Review, 6*.

Zhang, Y., Chen, Z., & Wei, B. (2020). *A sport athlete object tracking based on deep sort and yolo V4 in case of camera movement.* Paper presented at the 2020 IEEE 6th international conference on computer and communications (ICCC).

Zhou, D., Chen, G., & Xu, F. (2022). Application of Deep Learning Technology in Strength Training of Football Players and Field Line Detection of Football Robots. *Frontiers in Neurorobotics, 16*, 867028.