Multiple Object Tracking for Complex Motion Patterns

Zhengpeng Li, Member, IANEG, Yuhang Bai, Jun Hu, Bin Yang, and Xuange Liu*

Abstract—**While current tracking methods excel in following large objects with predictable movement, they face limitations in complex backgrounds, extensive object movement ranges, and scenarios involving rapid camera motion. Moreover, many existing tracking models heavily rely on scale-space transformation techniques for feature extraction, often leading to the loss of vital spatial information. To tackle these challenges, we introduce a novel model named multi-kernel layered aggregation and enhancement based-yolo, which stands out as a single-stage object tracking model. This model incorporates a multi-kernel context enhancement module to widen the receptive field and enhance the capture of global contextual information, thereby elevating tracking accuracy. Additionally, we have introduced a multi-link downsampling module to mitigate potential spatial information loss resulting from scale transformation. Furthermore, our approach employs a dual association process integrating Kalman filters and the Hungarian algorithm for both low and high-score detection boxes, effectively mitigating target loss caused by temporary detection failures. Experimental results on the SportsMOT dataset demon-strate that our model exhibits superior performance in tracking irregularly moving objects, especially in detecting and tracking small objects. It outperforms most existing object tracking models with a DetA score of 84.6 and a HOTA score of 68.5.**

Index Terms—**Object Detection, Multiple Object Tracking, Multi-Link Downsampling, Multi-Kernel Context Enhancement**

I. Introduction

In t recent years, multiple object tracking (MOT) [1] has garnered significant attention in the field of computer vision, with the simultaneous estimation of position and identity of different objects in camera-captured visual scenes posing a key challenge[2]. MOT has been extensively studied and applied in areas such as animal tracking[3], pedestrian behavior analysis[4], vehicle analysis[5], and driving scenarios[6],[7], [8]. However, its application in the more challenging sports scenes is still underdeveloped[9]. Tracking movement trajectories in sports allows coaches to deeply analyze their own team's tactical layout and gain insights into the opponent's strategies. Further data analysis can reveal which athletes need to increase training intensity or adjust their positions on the field to more effectively implement tactical strategies[10].

In deep learning-based MOT tasks, object detection networks and tracking algorithms are two crucial stages. In the object detection phase, scale-space transformation techniques like max pooling are widely employed to reduce image size. However, reliance solely on max pooling poses challenges. Specifically, since it only selects the maximum value within each window, other pixel value information is not considered in the computation, potentially leading to the loss of important spatial information or subtle features, thus impacting tracking accuracy.

The YOLO series of models are classic single-stage network models[11], with the SPP module first introduced in YOLOv3[12]. However, the SPP module falls short in extracting contextual information and spatial dimension information, especially in complex backgrounds and fast-moving objects. As sports scenes mainly involve small and medium-sized objects, specific solutions tailored to these targets are required to improve the performance of object detection networks.

The main contributions of this paper are as follows:

We propose a image-feature-free cross-drone multi-target association method. It remains robust in the face of unreliable image features and variable shooting angles. We propose a TMR-based re-association approach using cost evaluation that can further optimize the preliminary results. Extensive experiments on airsim-based dataset verify state-of-the-art performance of our proposed method.

(1) we introduce a single-stage object tracking model which named multi-kernel layered aggregation and enhancement based-yolo (MuKLYOLO).

(2) Introduction of a novel multi-link downsampling (MLD) module, which reduces the size of feature maps while retaining their most significant information, successfully addressing the challenges that max pooling might face in capturing complex spatial relationships and processing fixed structures.

Manuscript revised Sep 2, 2024; revised Dec 24, 2024.

The research work was supported by Fundamental Research Project of Higher Education Institutions by Liaoning Provincial Department of Education (LJ222410146057, LJ212410146040, and LJ212410146006), and Graduate Education Reform and Scientific and Technological Innovation and Entrepreneurship Project of University of Science and Technology Liaoning (LKDYC202403).

Zhengpeng Li and Yuhang Bai contributed equally to this work.

Zhengpeng Li is a doctoral student of University of Science and Technology Liaoning, Anshan, 114051, China. (e-mail: lkdlzp0901@163.com).

Yuhang Bai is a graduate student of University of Science and Technology Liaoning, 125105, China. (e-mail: 604760451@qq.com).

Jun Hu is a professor of University of Science and Technology Liaoning, Anshan, 114051, China. (e-mail: 320083700074@ustl.edu.cn)

Bin Yang is an associate professor of University of Science and Technology Liaoning, Anshan, 114051, China. (e-mail: yangbin673039297@126.com).

Xuange Liu is a teaching assistant at University of Science and Technology Liaoning, Anshan, 114051, China. (corresponding author to provide e-mail: liuxuangelxg@outlook.com).

(3) Design of a multi-kernel context enhancement (MKCE) module aimed at enhancing detection performance. This module dynamically adjusts the spatial receptive field, improving the network model's understanding of global contextual information. Depending on different detection scenarios, the network can adaptively adjust the perception range of the feature map, thereby obtaining more accurate detection results.

II. Related Work

Object tracking, an extension of the object detection task, constitutes a significant undertaking within the realms of computer vision and image processing. Object tracking is a critical area of research in computer vision, and numerous methodologies have been proposed according to different research focuses[13]. Among these, object tracking networks based on deep learning methods have gained popularity among researchers. In this approach, object detection serves as the foundation for multi-object tracking, with many prevalent models emerging from research based on object detection networks. Tracking networks based on the detection paradigm separate object detection and tracking into two distinct steps. They initially detect objects in video frames and then track these objects across successive frames to produce tracking results[14]. This method has gained widespread popularity in recent years. Based on the existing detection paradigm tracking networks, they can be categorized into networks based on single-stage models, dual-stage models, and Transformer network models. A multitude of methods focus on leveraging these given detection results to enhance tracking performance.

In the paradigm of single-stage network models for object tracking, Ma et al. [15] employed the K-means clustering algorithm to perform cluster analysis on candidate object boxes. They selected appropriate numbers of Anchor Boxes and added feature extraction layers in the shallow network layers to extract more refined vehicle features. This innovation improved the feature representation capability of the tracking model, though the model's effectiveness in handling occluded objects needs enhancement. FairMOT [16] seamlessly integrates object detection and re-identification tasks by sharing the backbone network and employing multi-task learning to achieve efficient multi-object tracking, demonstrating good adaptability to complex scenes. Wang et al. [17] proposed a hierarchical single-branch network built on a single-stage framework, which generates detection results and tracking features for objects in a single inference process. However, its tracking accuracy may be affected under extreme conditions, such as rapid movement or occlusion. Duan et al. [18] introduced CenterTrack, which predicts the center points of each object in consecutive frames and estimates the displacement between these centers to associate targets across frames, achieving object tracking. Yet, it faces challenges in handling prolonged occlusions. Generally, single-stage tracking algorithms are more lightweight and offer faster tracking speeds, making them highly regarded among researchers.

In the paradigm of dual-stage network models for object tracking, Fischer et al. [19] proposed QDTrack, a method that adopts a feature matching strategy. It computes and matches descriptors for almost every pixel in the image, not just for detected objects. This comprehensive feature analysis strategy enhances the tracking network's capability in handling occlusions and dynamic, fast-moving objects, though improvements are needed when dealing with objects of highly similar appearances. Chen et al. [20] introduced a tracking network that uses a Siamese network in the first stage for rough localization of objects, along with scale and ratio estimation. The second stage employs a segmentation network to precisely distinguish between the object and background, providing rotated bounding boxes for objects, which is effective for irregularly shaped or occluded objects. Leng et al. [21] proposed a template updating module to address the issue of tracking accuracy decline in SiamRPN when there are significant appearance changes in the object. This method updates the object in real-time during the second stage using this module to adapt to changes in the object's appearance. However, compared to single-stage models, dual-stage tracking frameworks are more computationally intensive in the field of visual object tracking.

In the paradigm of transformer-based network models, the Transformer architecture, widely applied in the field of deep learning, has been recently introduced to the domain of object tracking by researchers. Zhou et al. [22] proposed the GTR model, a network capable of directly processing multi-frame data to output object trajectories. This approach circumvents the complex trajectory association steps of traditional multi-object tracking methods, thus enhancing overall efficiency and accuracy. However, the model's limited feature extraction capability requires improvement in handling small and medium-sized objects. Nijhawan et al. [23] introduced the TransTrack model, ingeniously leveraging the Transformer architecture. It uses the object features from the previous frame as queries for the current frame and introduces a set of learned object queries (for newly appearing targets) to detect newly emerged objects. This method achieves object detection and association in a single process. Similar to dual-stage network models, Transformer-based models are also computationally intensive.

III. Methodology

Object tracking, an extension of the object detection task, constitutes a significant undertaking within the realms of computer vision and image processing. The MuKLYOLO-Byte network model adheres to a detect-track paradigm, as illustrated in Figure 1. In the detection phase, the input data initially passes through the backbone network, where preliminary extraction of top and middle layer features is performed using Focus, CBS, and residual modules. Lower layer features are then extracted through the MKCE module and MLD module. Subsequently, the neck of the model processes these features, facilitating the fusion of different hierarchical feature maps. This integration provides a richer and more distinctive blend of lower, middle, and upper layer information for subsequent object localization and classification tasks. The decoupled head handles the tasks of object detection localization and category classification separately. It localizes targets on a category-independent basis

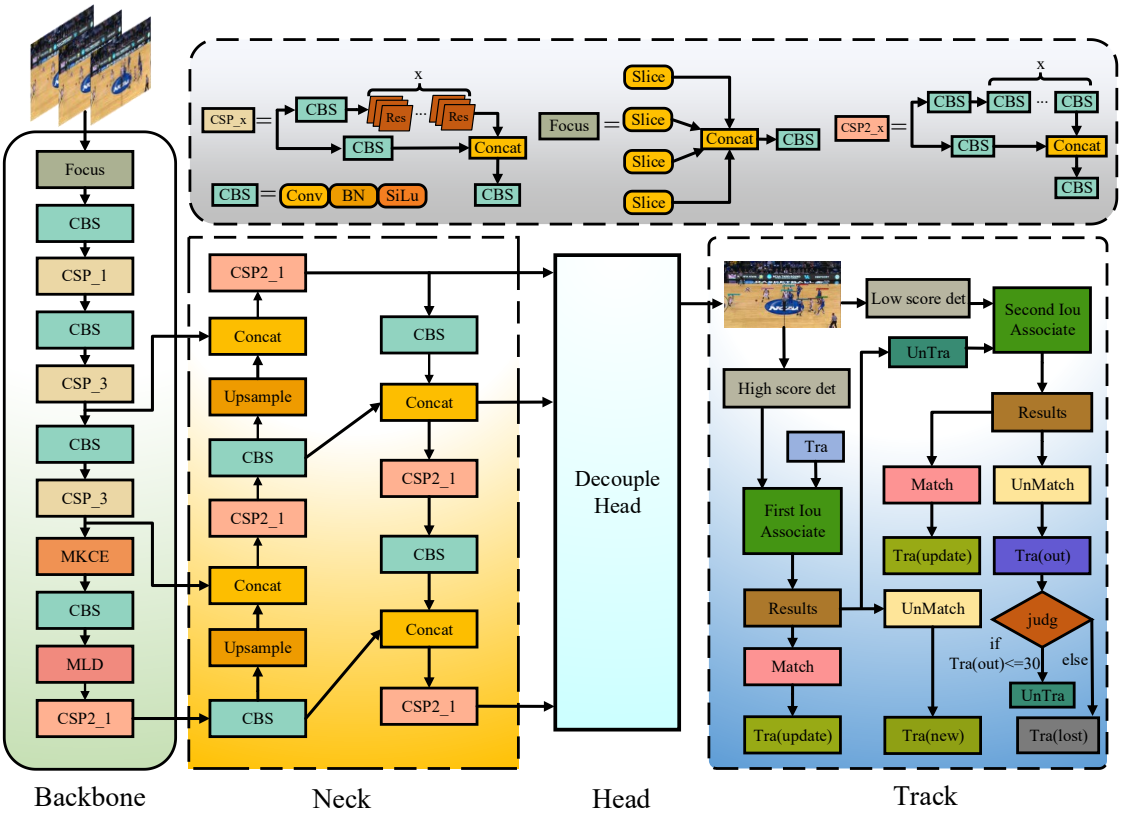

Fig. 1. Overview of the MuKLYOLO-Byte network.

and outputs categories through an independent network, thereby simplifying the detection task and enhancing detection performance. In the object tracking phase, we propose a combined optimization strategy using Kalman filters and the Hungarian algorithm to decide on the detection boxes.

A. Multi-link downsampling

Scale-space transformation techniques are widely utilized to reduce image size for extracting local feature information. However, this approach has a notable drawback: the loss of detail information, which adversely affects the detection of small and medium-sized objects. In various ball sports, the required amount of contextual information varies depending on the characteristics of the game scene.

For instance, in volleyball and basketball games, due to the relatively small camera angles and players being distributed around various corners of the camera's field of view, global contextual information becomes particularly crucial. In soccer matches, where the field is larger and the camera angle more extensive, with concurrently smaller target sizes, acquiring and analyzing global contextual information becomes even more vital. This demonstrates the prominent role of remote modeling in feature content within ball game scenarios.

The backbone network of YOLOX [24] includes the Focus module, CBS module, CSPLayer compound residual module, and the SPP module. The SPP module, as shown in Figure 2, employs multiple scale max-pooling layers to capture contextual information for each pathway, then flattens the results and concatenates them into a feature vector. For small objects, SPP can lead to the dilution or loss of important features. Even with pooling at various scales, key features of small objects can become blurred. Therefore, this paper proposes a MLD module, as depicted in Figure 3, to address these issues.

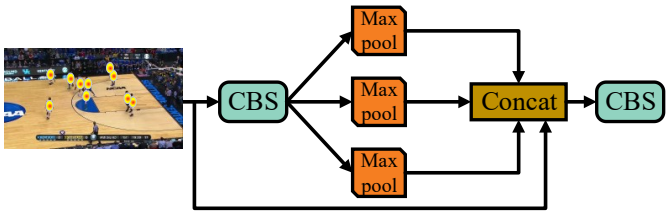

Fig. 2. Overview of the SPP.

As the feature map passes through this module, it initially traverses the CBS (convolution, batch normalization, and Silu) module. Subsequently, it flows through three main pathways, utilizing convolutional kernels of sizes 5×5, 9×9, and 13×13, respectively named as *main*1, *main*2, and *main*3. Each main pathway is further divided into three sub-pathways, named *branch*1, *branch*2, and *branch*3. To achieve optimal results, unique feature extraction methods are employed for each branch to derive their outcomes. The results from the three sub-pathways are then used to compute the output of each main pathway. Ultimately, the outputs of the three main pathways are concatenated to form a channel re-calibrated feature map, which once again passes through the CBS module, resulting in the output of the multi-link downsampling process. The three sub-pathways are central to channel re-calibration, enabling them to learn different channel information based on their respective convolutional

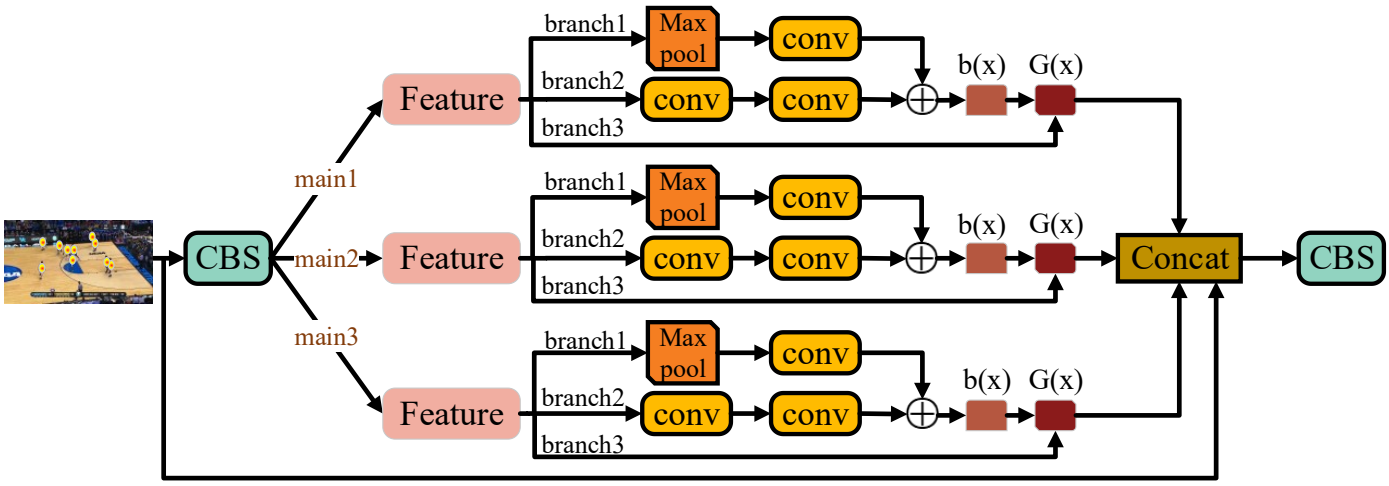

Fig. 3. Overview of the MLD.

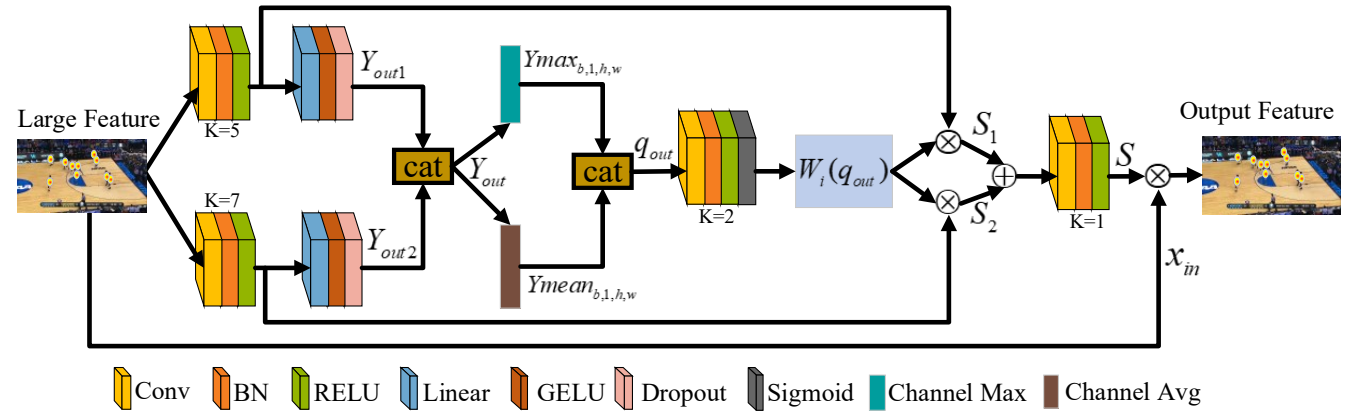

Fig. 4. Overview of the MKCE.

kernel sizes. The specific derivation process for each main pathway is as follows:

Initially, this module processes the feature outputs from two sub-pathways. For an input feature map x, $branch1$ utilizes a concatenation of max pooling and convolution to generate its output, as illustrated in equations (1) and (2). In these equations, i and j denote the coordinates in the feature map F, while m and n represent the local coordinates within the pooling window. $p = \lfloor k/2 \rfloor$ signifies the stride, with $s = 1$ indicating the stride of the padding, and k representing the size of the convolutional kernel.

$$M([i,j]) = \max_{0 \le h < k, 0 \le w < k} (x[i \times s : i \times s + h - p, \; j \times s : j \times s + w - p]) \quad (1)$$

where M indicates the output result after the maxpool.

$$b_1(x) = \delta(BN(conv(M(x)))) \quad (2)$$

where δ denotes the ReLU activation function, and BN signifies the batch normalization operation.

The second sub-pathway, denoted as $b_2(x)$, establishes a dual convolutional structure to extract feature information, as indicated in equation (3). Subsequently, the enhanced semantic information is obtained through an addition operation of $b_1(x)$ and $b_2(x)$, as illustrated in equation (4).

$$b_2(x) = conv(conv(x)) \quad (3)$$

$$b(x) = b_1(x) + b_2(x) \quad (4)$$

Finally, to preserve the original spatial information, the module ensures that primary details are retained even when the feature is processed in deeper layers of the network. Therefore, a residual connection approach is utilized. The input feature x is residually fused with the output of $b(x)$. This results in the production of an enhanced sub-pathway summarized feature map output $G(x)$, demonstrating the effectiveness of the residual fusion in maintaining spatial details.

$$G(x) = x + b(x) \quad (5)$$

In summary, the MLD module employs a multi-sub-pathway design in a strategic manner, integrating convolution and residual connections to capture a diverse array of local information. This approach effectively mitigates the issue of detail loss that may arise from max pooling. Convolution operations are utilized to capture advanced local features within each channel, such as edges and detailed information, while max pooling reduces the spatial dimensions of the data, retaining key features. When processing multi-channel inputs,

the combined use of different sizes of convolutional kernels for pooling and convolution operations offers certain advantages, as it allows for complex, multi-level integration of feature information. Integrating this module enhances comprehensive feature extraction and bolsters the model's understanding of complex data structures. Specific ablation study analyses are presented in Table 4.

B. Multi-kernel context enhancement

The MKCE structure is depicted in Figure 4.4. The MKCE module is composed of a feature linear layer, a dual aggregation layer, and a feature recalibration layer. Initially, the feature linear layer uses convolutional kernels of varying sizes on a large feature map to capture multi-scale local features. Small kernels focus on extracting fine-grained features, while large kernels emphasize broad contextual information. Subsequently, the dual aggregation layer models the concatenated feature map along the channel direction, aiming to capture the intricate details and contextual relationships between channels. In this process, MKCE calculates both the maximum and average value weights. These weights are then used to effectively integrate the relevant information into the local feature map. Finally, the feature recalibration layer merges the weighted features with the initial features, resulting in a more refined and comprehensive global feature map (Output Feature). The specific computational process is as follows:

In the first step, the feature linear layer employs an expanded convolution sequence with larger convolution kernels and increased dilation coefficients to construct a more extensive kernel convolution sequence. Specifically, for the $m-th$ depth convolution in the sequence, the kernel size k, dilation parameter d, and receptive field RF are defined as shown in equations (6).

$$RF_m = RF_{m-1} + d_m(k_m - 1) \tag{6}$$

where $k_{m-1} \le k_m$, $d_1 = 1$, $d_{m-1} < d_m \le RF_{m-1}$, $RF_1 = k_1$. Two depth convolutions, $Conv1$ and $Conv2$, are selected, and a linear transformation is added after each convolution to capture richer contextual relationships within the feature map, as indicated in equations (7) and (8).

$$Y_{out1} = Drp(GELU(LN(Conv1(x_{in})))) \tag{7}$$

$$Y_{out2} = Drp(GELU(LN(Conv2(x_{in})))) \tag{8}$$

where Drp represents the dropout operation, and LN stands for the linear layer. The parameters for the two depth convolutions ($Conv1$ and $Conv2$) are set as $d1 = 1$, $d2 = 3$, $k1 = 5$, $k2 = 7$, $p1 = 2$, $p2 = 9$, with Y_{in}, Y_{out1}, and Y_{out2} representing the input and outputs, respectively.

To assimilate a broader range of channel information, the feature linear layer concatenates the results from different convolution kernels, as demonstrated in equation (9).

$$Y_{out} = cat(Y_{out1}, Y_{out2}) \tag{9}$$

where cat denotes the concatenation operation.

In the second step, the information aggregation layer merges the central tendency and peak characteristics of the data, thereby aiding in enhancing the learning efficacy. This layer utilizes spatial channel information to compute the mean $Ymean$ and maximum $Ymax$ values along the channel dimension, capturing the overall characteristics and most distinctive features of the data, as shown in equations (10) and (11).

$$Ymean_{b,1,h,w} = \frac{1}{C}\sum_{i=1}^{C} Y_{b,i,h,w} \tag{10}$$

$$Ymax_{b,1,h,w} = \max_c Y_{b,c,h,w} \tag{11}$$

where Y represents the input tensor, and C denotes the number of channels.

The enhanced output q_{out} is obtained by concatenating $Ymean$ and $Ymax$ along the first dimension, as indicated in equation (12). The process involves further information extraction from q_{out} using convolution to enhance the inter-channel relationships, as seen in equation (4.13).

$$q_{out} = cat[Ymean_{b,1,h,w}, Ymax_{b,1,h,w}] \tag{12}$$

$$W_i(q_{out}) = \sigma(ReLu(BN(conv(q_{out})))) \tag{13}$$

where $W_i(q_{out})$ represents the different selection masks for two channels, σ signifies the sigmoid activation function, the convolution kernel size is $k = 2$, and i represents different channels.

In the third step, within the feature recalibration layer, the outputs from the two deep convolutions ($Conv_i$) of the feature linear layer are multiplied by the different channel weights $W_i(q_{out})$ obtained from the information aggregation layer. This is demonstrated in equation (14).

$$S_i = W_i(q_{out}) \times Conv_i \tag{14}$$

where $i \in \{1, 2\}$.

Subsequently, the resulting Si is concatenated, and convolution is used to increase the dimensionality of the feature, resulting in the output S, as shown in equation (15).

$$S = ReLu(BN(Conv(cat[S_i]))) \tag{15}$$

Finally, S is multiplied with the original input to produce the weighted output, as shown in equation (16).

$$O = x_{in} \times S \tag{16}$$

C. Object tracker

In traditional object tracking strategies, models often rely on detection boxes with high confidence for object association and tracking, as these high-confidence boxes are more likely to represent actual objects and are therefore given higher priority. However, detection boxes with low confidence might represent false detections but could also indicate real objects. Merely ignoring or discarding these low-confidence boxes can lead to missed real targets. The MuKLYOLO-Byte tracker, inspired by the BYTE [25] algorithm, adopts a different approach, utilizing dual association through Kalman filters and the Hungarian algorithm, integrating low-confidence detection boxes for association matching. This algorithm enhances the overall tracking performance. However, target tracking networks based on the detect-track paradigm still require high-performance object detection networks to reduce the number of undetected negatives.

Unlike most object tracking algorithms, which traditionally discard low-scoring prediction boxes, our method retains

nearly all detection boxes, dividing them into high-scoring and low-scoring groups. As a result, targets are not prematurely ignored due to low detection scores, even in cases of occlusion, motion blur, or changes in target size.

The MuKLYOLO-Byte object tracker, as illustrated in Figure 1 (Track), takes an object detector and a video sequence as inputs and sets two thresholds, τ=0.40 and η=0.10. By comparing these thresholds with the confidence p of the detector's output, detection boxes are classified: boxes with $\tau \le p$ are categorized as high-confidence, and those with $\eta \le p \le \tau$ as low-confidence, while boxes with $p \le \eta$ are discarded.

The object tracker involves a two-step association process [19]. After classification through thresholds, the first step involves associating high-score detection boxes with current track fragments. If some track fragments fail to match with high-score boxes, they are then associated with low-score boxes. This dual association process aims to prevent target loss due to temporary detection failures. Specific analyses of these two tracking steps are as follows:

In the first association, Kalman filters are used to predict the positions of each high-confidence detection box in the current frame, marked as $Tra(update)$. Detection boxes that are not associated are set as new trajectories, $Tra(new)$. During this process, the intersection over union (IOU) between high-confidence boxes and trajectories is calculated to assess the association degree. The Hungarian algorithm is then utilized to match similarities and achieve the optimal "detection box-trajectory" correspondence. Finally, unmatched track pools, $UnTra$, are retained.

In the second association, low-confidence detection boxes are matched with trajectories marked as UnTra. To maintain long-term association consistency, all track pools not matched in the second association, $Tra(out)$, are retained, with each trajectory's duration limited to 30 frames. If a trajectory reappears in $Tra(out)$ in subsequent associations, it will be rematched; if it does not appear within the specified number of frames, the trajectory is deleted.

D. Loss Function

The loss function of MuKLYOLO-Byte comprises coordinate loss, confidence loss, and classification loss. The coordinate loss employs the intersection over union (IoU) loss function, while the confidence loss and classification loss use the cross-entropy loss function.

For a more accurate measurement of the overlap between predicted and actual bounding boxes, Generalized IoU Loss (GIoULoss) is utilized, represented as L_{bbox} in equation (17).

$$L_{bbox}(A_1, A_2) = \frac{|I|}{|U|} - \frac{|C-U|}{|C|} \tag{17}$$

where A_1 and A_2 are two bounding boxes, I is their intersection, U is their union, and C denotes the smallest enclosing rectangle containing both A_1 and A_2.

The classification loss primarily evaluates the discrepancy between the predicted class probabilities of the model and the actual classes. For this purpose, the cross-entropy loss function is employed, represented as L_{class} in equation (18).

$$L_{cls} = -\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{C} y_{ij}\log(\hat{y}_{ij}) \tag{18}$$

where N indicates the total number of samples, and C the number of classes. y_{ij} is the one-hot encoding of the true class j for the $i-th$ sample, where the specific class value is 1, and all other class values are 0. $\hat{y}_{ij}$ represents the model's predicted probability that the $i-th$ sample belongs to class j.

The confidence loss is typically used to measure the discrepancy between the model's predicted confidence and the actual confidence, for which the binary cross-entropy loss function is applied, represented as L_{obj} in equation (19).

$$L_{obj} = -\frac{1}{N}\sum_{i=1}^{N}(z_i\log(\hat{z}_i) + (1-z_i)\log(1-\hat{z}_i)) \tag{19}$$

where z_i is the actual confidence of the $i-th$ sample, and $\hat{z}_i$ is the model's predicted probability of the existence of the $i-th$ sample.

The overall loss function of the model is the sum of coordinate loss, confidence loss, and classification loss, as shown in equation (20).

$$Loss = L_{bbox} + L_{cls} + L_{obj} \tag{20}$$

IV. Experiments

A. Evaluation Index

Evaluation metrics in deep learning play a pivotal role in research and applications. They quantify model performance and provide researchers with an objective and comparable means of assessment. Currently, multi-object tracking primarily employs metrics such as HOTA, MOTA, IDs, AssA, DetA, and Frag, which are also adopted in this study.

The IDs metric primarily reflects the performance of an object tracking network in terms of continuity and association accuracy. A lower IDs value indicates better continuity and association.

AssA is an assessment metric for multi-object tracking performance. This metric involves two main types of error scenarios: 1) Predicted targets may be incorrectly associated with non-corresponding real targets; 2) A real target may be erroneously segmented into multiple predicted outcomes. Through such quantification, AssA accurately evaluates the algorithm's performance in target association. The specific calculation steps can be referred to in equations (21), (22), and (23).

$$\alpha(c) = \frac{|TPA_C|}{|TPA_C + FNA_C + FPA_C|} \tag{21}$$

$$AssA_\beta = \frac{1}{|TP|}\sum\nolimits_{c\in\{TP\}}\alpha(c) \tag{22}$$

$$AssA = \int_0^1 AssA_\beta d_\beta \tag{23}$$

where $\alpha(c)$ represents the probability of measuring the association between predicted trajectories and actual label

trajectories, with $\beta \in \{0.05, 0.1...0.9, 0.95\}$ denoting the baseline for localization similarity. FNA_c represents the number of instances where some real targets are not correctly associated with predicted targets, TPA_c indicates the number of correct associations between predicted and real targets, and FPA_c denotes the number of instances where predicted targets are wrongly associated with other real targets.

The DetA metric evaluates the accuracy of target detection in multi-object tracking networks. Specifically, it quantifies the degree of match between the targets detected by the network and the actual targets. DetA integrates recall and precision to comprehensively assess the network's detection performance. The calculation process is as shown in equation (24) and (25).

$$DetA_\beta = \frac{|TP|}{|TP| + |FN| + |FP|} \tag{24}$$

$$DetA = \int_0^1 DetA_\beta d_\beta \tag{25}$$

where FN represents false negatives, FP denotes false positives, and TP stands for true positives.

Multiple object tracking accuracy (MOTA) [26] is a critical metric for assessing the accuracy of multi-object tracking algorithms. It provides an overall performance score for tracking algorithms by integrating false positives, missed detections, and identity switches. The MOTA metric intuitively expresses the overall effectiveness of a tracking algorithm, as illustrated in equation (26). At $i-th$ time, a_i represents the number of missed detections, fp_i the number of false positives, ae_i the number of incorrect matches, and gt the number of true targets.

$$MOTA = 1 - \frac{\sum_i (a_i + fp_i + ae_t)}{\sum_i gt_i} \tag{26}$$

Higher order tracking accuracy (HOTA) [27] is an evaluation metric that measures the overall performance of object tracking networks. It integrates the accuracy of both detection and association into a single metric. Compared to other metrics, HOTA provides the most comprehensive assessment of tracking network performance and is currently the most accurate metric for evaluating object tracking. The calculation process of HOTA is demonstrated in equations (27) and (28).

$$HOTA_\beta = \sqrt{\frac{\sum_{c \in \{TP\}} \alpha(c)}{|TP| + |FN| + |FP|}} \tag{27}$$

$$HOTA = \int_0^1 HOTA_\beta d_\beta \approx \frac{1}{19} \sum_\beta HOTA_\beta \tag{28}$$

Frag is a metric used to quantify the performance of multi-object tracking networks. It evaluates the continuity and consistency of an object trajectory over time. Specifically, it calculates the frequency of state transitions between an object trajectory being successfully tracked and losing tracking. This is represented in equation (29),

$$Frag = \sum_{i=1}^{n} |t_i - t_{i-1}| \tag{29}$$

where n denotes the total number of frames and $t_i \in \{0,1\}$ indicates the state of an object being successfully tracked in frame i.

B. Experiment setting

The MuKLYOLO-Byte network model was designed and implemented using the Pytorch deep learning framework. Our training and testing were conducted on an A100 graphics card. We utilized the stochastic gradient descent algorithm to optimize training parameters, with a momentum parameter set to 0.90 and a weight decay parameter of 5e-04. The maximum epoch was set to 40, with an initial learning rate of 3.0e-05. In the final 10 epochs, the learning rate was reduced to 50% of its initial value. The MuKLYOLO-Byte network model applied various image enhancement techniques, including image cropping, random flipping, mix-up augmentation, Mosaic, and RandomAffine, and set the training input image dimensions accordingly.

C. Baseline model

We selected six state-of-the-art (SOTA) models for baseline comparison with the MuKLYOLO-Byte model, detailed as follows:

CenterTrack [18]: This method innovates in the detection aspect, not by simply detecting objects independently in each frame, but by tracking the center points of objects. This approach reduces the computational load without compromising detection performance. For the tracking aspect, CenterTrack learns the 2D offset between two adjacent frames to predict the next frame.

ByteTrack [25]: This method proposes a simple, effective, and universal multi-object tracking method. Its core idea is to associate almost every detection box in the video, not just the high-scoring ones. For low-scoring detection boxes, it uses their similarity to trajectories to recover true objects and filter out the background.

GTR [22]: This method introduces a Transformer-based global multi-object tracking network model. In object detection, it accepts continuous video frames to encode object features and effectively integrates these features into complete trajectories using trajectory querying technology. In object tracking, GTR introduces a global tracking module that operates over the entire video sequence to ensure continuous and accurate object tracking.

MixSort-Byte [28]: This Transformer-based object tracking network optimizes object association by generating a mixed similarity matrix. This association strategy not only considers the appearance features of objects but also their motion trajectories, thereby achieving more accurate multi-object tracking.

FairMOT [16]: The FairMOT multi-object tracking method differs from traditional methods that treat object detection and re-identification as two independent tasks. Based on an anchor-free object detection architecture, FairMOT achieves a balanced integration of detection and re-ID within the same network, resolving the competition between detection and re-ID and ensuring balance in a single network.

Fig. 5. Sample images from the SportsMOT dataset of basketball, volleyball, and soccer.

QDTrack [19]: This network employs a "Quasi-Dense Similarity Learning" strategy, densely sampling and conducting contrastive learning in object regions of an image. This strategy utilizes most information areas within the image to enhance tracking accuracy.

D. Dateset

To validate the effectiveness of MuKLYOLO-Byte, we utilized the large-scale SportsMOT [28] dataset. This dataset comprises a total of 240 video sequences, including 45 training videos, 45 validation videos, and 150 test videos. It encompasses videos of three types of sports: basketball, volleyball, and soccer.

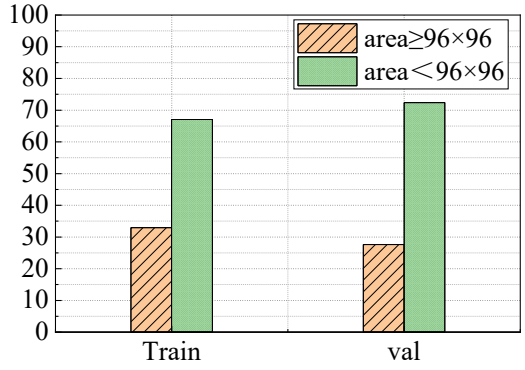

Fig. 6. Proportional division of the SportsMOT dataset.

As shown in Table 1, the dataset details the average number of frames per video category, the average number of detection boxes per frame, the number of tracking IDs (Track IDs), and the track gap length. The SportsMOT dataset presents three major challenges. Due to the dynamic nature of sports scenarios, object association in multi-object tracking becomes complex. The fast and variable-speed movements in sports scenarios pose challenges to trackers. Although MOT trackers primarily rely on the appearance of objects, similar uniforms make differentiation difficult. As illustrated in Figure 5, the first row of images shows the challenges of dynamic scenes, the second row displays challenges due to fast and variable-speed movements, and the third row reveals the issue of similar appearances among team members.

We categorized the detection boxes in the training and validation sets of the SportsMOT dataset into large, medium, and small targets. Specifically, large targets have an area greater than 96x96 pixels, while medium and small targets range from 0 to 96x96 pixels in area. Following this criterion, we also classified the targets in the SportsMOT dataset accordingly. As depicted in Figure 6, it is evident that medium and small detection targets constitute over 60% of the total in both the training and validation sets.

Table 1
Detailed statistics of the three categories in the SportsMOT dataset.

Labels	Frames	Track IDs	Bboxs	Track gap len
soccer	845	10	9	69
basketball	360	12	11	38
volleyball	674	21	13	116

V. Discussion

A. Contrast Experiment

To validate the effectiveness of the MuKLYOLO-Byte, various recent tracking paradigm models were selected as baselines for comparison, with experiments conducted on the official SportsMOT dataset test set. The comparative results are presented in Table 2, where a downward arrow (↓) indicates that lower scores denote better performance, an upward arrow (↑) signifies that higher scores indicate better performance, and bold text highlights the best performance.

Compared to FairMOT, CenterTrack, and ByteTrack, which employ detect-associate single-stage object tracking methods, the MuKLYOLO-Byte achieved improvements of 19.2 points, 5.8 points, and 4.4 points, respectively, in the HOTA metric. In the DetA metric, it showed improvements of 14.4 points, 2.5 points, and 6.0 points, respectively, and in the AssA metric, it improved by 11.9 points, 7.6 points, and 3.3 points, respectively.

Table 2
Comparative table of the MuKLYOLO-Byte model, where a downward arrow (↓) indicates that a lower metric is better, an upward arrow (↑) signifies that a higher metric is better, and bold text represents the best performance.

	HOTA↑	MOTA↑	AssA↑	IDs↓	DetA↑	Frag↓
FairMOT[16]	49.3	86.4	43.7	9928	70.2	21673
GTR[22]	54.5	67.9	45.9	9567	64.8	14525
QDTrack[19]	60.4	90.1	47.2	6377	77.5	11850
CenterTrack[18]	62.7	90.8	48.0	10481	82.1	5750
ByteTrack(YOLOX-s) [25]	64.1	95.9	52.3	3089	78.6	4216
MixSort-Byte[28]	65.7	**96.2**	54.8	**2472**	78.8	**4009**
MuKLYOLO-Byte (ours)	**68.5**	93.5	**55.6**	3548	**84.6**	4020

Table 3
Object detection results of the MuKLYOLO-Byte, using the validation set of the SportsMOT dataset for experimental results.

	MAP↑	HOTA↑	APLarge↑	APmedium↑	APSmall↑
ByteTrack(YOLOX-s)	97.4	70.3	98.5	97.4	86.7
MuKLYOLO-Byte (ours)	97.7	74.0	98.7	97.8	89.0

Table 4
Sub-pathway ablation experiment table, where a check mark (√) indicates the inclusion of a specific computational scheme, and a dash (-) denotes its absence.

branch1	branch2	branch3	HOTA↑	DetA↑
-	-	-	70.3	83.9
√	-	-	71.3	86.2
-	√	-	71.9	86.1
-	-	√	71.6	85.9
√	√	-	72.3	86.4
-	√	√	72.7	86.3
√	-	√	72.5	86.3
√	√	√	**73.4**	**86.6**

Fig. 7. Visualization results for basketball tracking.

Fig. 8. Visualization results for volleyball tracking.

Fig. 9. Visualization results for soccer tracking.

Table 5
Ablation experiment table for the Multi-Kernel Context Enhancement module.

Max	Avg	branch1	branch2	branch3	HOTA↑	DetA↑
√	-	√	√	√	72.6	86.2
-	√	√	√	√	72.4	86.3
√	√	√	√	√	**74.0**	**87.1**

Against QDTrack, which uses a detect-embed dual-stage object tracking method, the MuKLYOLO-Byte improved by 8.1 points in HOTA, 7.1 points in DetA, and 8.4 points in AssA. When compared with Transformer-based tracking models GTR and MixSort-Byte, the MuKLYOLO-Byte showed improvements of 14.0 points and 2.8 points in HOTA, 19.8 points and 5.8 points in DetA, and 9.7 points and 0.8 points in AssA, respectively. These results demonstrate that the MuKLYOLO-Byte network model holds competitive performance amongst recent mainstream object tracking networks.

When compared with Transformer-based tracking models GTR and MixSort-Byte, the MuKLYOLO-Byte showed improvements of 14.0 points and 2.8 points in HOTA, 19.8 points and 5.8 points in DetA, and 9.7 points and 0.8 points in AssA, respectively. These results demonstrate that the MuKLYOLO-Byte network model holds competitive performance amongst recent mainstream object tracking networks.

The MuKLYOLO-Byte model has demonstrated improvements in object detection performance, subsequently enhancing tracking performance. This section presents an experimental analysis of the model's detection effectiveness, as detailed in Table 3. Since the SportsMOT dataset's test set is not publicly available, the validation set was used for the experiments. The experimental results of MuKLYOLO-Byte indicate a significant improvement in object detection performance compared to the ByteTrack model. MuKLYOLO-Byte exhibited superior capabilities in handling targets of various sizes. The improvement in the mAP (mean Average Precision) metric highlights a clear enhancement in the model's overall detection accuracy. The advancements in the APLarge, APMedium, and APSmall metrics, which respectively target large, medium, and small-sized objects, further demonstrate MuKLYOLO-Byte's superiority in handling targets of varying sizes.

Overall, MuKLYOLO-Byte not only shows an improvement in overall object detection accuracy but also displays a more balanced and efficient performance in handling targets of different sizes. This enables the model to provide more reliable and precise object detection results in complex sports scenarios.

B. Data Visualization

In basketball and volleyball videos, medium-sized targets are commonly seen within the camera's perspective, often exhibiting rapid movements coupled with occlusion challenges, posing difficulties for object tracking networks. Soccer videos typically feature larger scenes with an abundance of small targets. This demands that the network model adapt appropriately to targets of various sizes while ensuring tracking accuracy. The MuKLYOLO-Byte network model is capable of high-quality tracking of each target in scenarios involving rapid movement and occlusions. This section presents visual output results to showcase the network's exceptional performance, as illustrated in Figures 7, 8, and 9, depicting visualization results for different sports categories.

As observed from the figures, basketball movements are complex with frequent occlusions, and MuKLYOLO-Byte effectively handles such challenges, accurately detecting targets and ensuring smooth ID transitions. Volleyball players exhibit rapid movements and often face occlusions. MuKLYOLO-Byte performs excellently in handling these fast-paced and occluded scenarios. In soccer videos, targets are farther away and smaller in size. MuKLYOLO-Byte accurately detects target positions in these scenes while minimizing the number of ID switches.

C. *Ablation Experiment*

We conducted ablation experiments on the MuKLYOLO-Byte network using the validation set of the SportsMOT dataset to demonstrate the effectiveness of each module. The setup for other parameters remained the same as in the training configuration.

More sub-pathways lead to better performance. Therefore, the ablation experiment was divided into different ways of adding sub-pathways to demonstrate how the choice of different pathways in the multi-link downsampling module can affect the overall performance of the model. This ablation experiment did not include the MKCE module. As shown in Table 4, the first three rows, which only contain a single pathway, perform the worst. The fourth, fifth, and sixth rows show improved performance with different pathway addition methods. The best result was obtained when the maximum number of pathways was used, as in row seven, where the HOTA and DetA metrics improved by 3.4 and 3.3 percentage points, respectively, compared to the first row.

The MKCE module uses an information aggregation layer to obtain unique information from feature maps. The channel-level maximum (Max) and average (Avg) values focus on different aspects of feature information. Choosing a reasonable method for information learning can enhance the overall performance of the model. For this reason, in the ablation experiment, we tested three schemes: using only Max, using only Avg, and using both. As shown in Table 5, the model achieved the best performance when using both channel average and channel maximum values, with the HOTA and DetA metrics improving by 1.6 and 1.2 percentage points, respectively, compared to the previous setup.

VI. Conclusion

We introduce the MuKLYOLO-Byte model, designed to address the issue of insufficient performance in current object tracking methods when dealing with rapidly moving small and medium-sized targets. The integration of the multi-link downsampling module aims to reduce the size of feature maps while preserving detailed information, effectively solving the problem of spatial information loss caused by scale-space transformation techniques. Additionally, the multi-kernel context enhancement module is employed. During the feature extraction process, this module dynamically adjusts the spatial receptive field to enhance the processing capability for global contextual information. This enables the network to adaptively adjust the perception field of the feature map, improving tracking accuracy for irregularly moving objects. Finally, the model utilizes a dual-association approach combining Kalman filters and the Hungarian algorithm for optimal selection of detection boxes. In evaluations on the SportsMOT dataset, the MuKLYOLO-Byte model demonstrates excellent performance. In future research, we plan to combine and innovate with existing object tracking methods to enhance accurate prediction of irregular target trajectories. We aim to further improve the model's applicability and generalization

capability, ensuring its robust performance in various complex scenarios.

REFERENCES

[1] W. W. Y. Ng, X. Liu, X. Yan, X. Tian, C. Zhong, and S. Kwong, "Multi-object tracking for horse racing," *Information Sciences,* vol. 638, 2023.

[2] L. Jiao, D. Wang, Y. Bai, P. Chen, and F. Liu, "Deep Learning in Visual Tracking: A Review," *IEEE Transactions on Neural Networks and Learning Systems,* vol. 34, no. 9, pp. 5497-5516, 2023.

[3] L. Zhang, J. Gao, Z. Xiao, and H. Fan, "AnimalTrack: A Benchmark for Multi-Animal Tracking in the Wild," *International Journal of Computer Vision,* vol. 131, no. 2, pp. 496-513, 2023.

[4] W. Do, N. Saunier, and L. Miranda-Moreno, "An empirical analysis of the effect of pedestrian signal countdown timer on driver behavior at signalized intersections," *Accident Analysis and Prevention,* vol. 180, 2023.

[5] I. Bisio, H. Haleem, C. Garibotto, F. Lavagetto, and A. Sciarrone, "Performance Evaluation and Analysis of Drone-Based Vehicle Detection Techniques From Deep Learning Perspective," *IEEE Internet of Things Journal,* vol. 9, no. 13, pp. 10920-10935, 2022.

[6] S. Zangenehpour, L. F. Miranda-Moreno, and N. Saunier, "Automated classification based on video data at intersections with heavy pedestrian and bicycle traffic: Methodology and application," *Transportation Research Part C: Emerging Technologies,* vol. 56, pp. 161-176, 2015.

[7] K. K. Santhosh, D. P. Dogra, P. P. Roy, and B. B. Chaudhuri, "Trajectory-Based Scene Understanding Using Dirichlet Process Mixture Model," *IEEE Transactions on Cybernetics,* vol. 51, no. 8, pp. 4148-4161, 2021.

[8] R. Wang *et al.*, "TIF: Trajectory and Information Flow Coupling Mechanism for Behavior Analysis in Autonomous Driving," *IEEE Transactions on Intelligent Transportation Systems,* vol. 23, no. 12, pp. 25216-25225, 2022.

[9] C.-Y. Chong, "An Overview of Machine Learning Methods for Multiple Target Tracking," in *24th IEEE International Conference on Information Fusion, FUSION 2021, November 1, 2021 - November 4, 2021*, Sun City, South africa, 2021: Institute of Electrical and Electronics Engineers Inc., p. Armscor; Mankwe Game Trackers; Metron; Multichoice Group; SA Tourism; Sun City.

[10] H. Agrawal, A. Halder, and P. Chattopadhyay, "A systematic survey on recent deep learning-based approaches to multi-object tracking," 2023.

[11] G. Zhang, W. Kang, R. Ma, and L. Zhang, "Multi-object Tracking Based on YOLOX and DeepSORT Algorithm," in *5th EAI International Conference on 6G for Future Wireless Networks, 6GN 2022, December 17, 2022 - December 18, 2022*, Harbin, China, 2023, vol. 504 LNICST: Springer Science and Business Media Deutschland GmbH, pp. 52-64.

[12] T. Mostafa, S. J. Chowdhury, M. K. Rhaman, and M. G. R. Alam, "Occluded Object Detection for Autonomous Vehicles Employing YOLOv5, YOLOX and Faster R-CNN," in *13th IEEE Annual Information Technology, Electronics and Mobile Communication Conference, IEMCON 2022, October 12, 2022 - October 15, 2022*, Virtual, Online, Canada, 2022: Institute of Electrical and Electronics Engineers Inc., pp. 405-410.

[13] Y. Xu, X. Zhou, S. Chen, and F. Li, "Deep learning for multiple object tracking: A survey," *IET Computer Vision,* vol. 13, no. 4, pp. 411-419, 2019.

[14] D. Wu, "Research on target tracking method of sports video based on multi-template matching," in *2020 International Conference on Virtual Reality and Intelligent Systems, ICVRIS 2020, July 18, 2020 - July 19, 2020*, Zhangjiajie, China, 2020: Institute of Electrical and Electronics Engineers Inc., pp. 82-85.

[15] Y.-J. Ma, Y.-T. Ma, S.-S. Cheng, and Y.-D. Ma, "Road vehicle detection method based on improved YOLO v3 model and deep-SORT algorithm," *Jiaotong Yunshu Gongcheng Xuebao/Journal of Traffic and Transportation Engineering,* vol. 21, no. 2, pp. 222-231, 2021.

[16] Y. He, J. Che, and J. Wu, "Pedestrian Multi-object Tracking Based on ResNeXt and FairMOT," in *3rd International Symposium on Automation, Mechanical and Design Engineering, SAMDE 2022, December 16, 2022 - December 18, 2022*, Beijing, China, 2023, vol. 138: Springer Science and Business Media B.V., pp. 193-205.

[17] F. Wang, L. Luo, E. Zhu, and S. Wang, "Multi-object Tracking with a Hierarchical Single-Branch Network," in *28th International Conference on MultiMedia Modeling, MMM 2022, June 6, 2022 - June 10, 2022*, Phu Quoc, Viet nam, 2022, vol. 13142 LNCS: Springer Science and Business Media Deutschland GmbH, pp. 73-83.

[18] K. Duan, S. Bai, L. Xie, H. Qi, Q. Huang, and Q. Tian, "CenterNet: Keypoint triplets for object detection," in *17th IEEE/CVF International Conference on Computer Vision, ICCV 2019, October 27, 2019 - November 2, 2019*, Seoul, Korea, Republic of, 2019, vol. 2019-October: Institute of Electrical and Electronics Engineers Inc., pp. 6568-6577.

[19] T. Fischer *et al.*, "QDTrack: Quasi-Dense Similarity Learning for Appearance-Only Multiple Object Tracking," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* pp. 1-13, 2023.

[20] F. Chen, F. Zhang, and X. Wang, "Two Stages for Visual Object Tracking," in *2021 International Conference on Intelligent Computing, Automation and Applications, ICAA 2021, June 25, 2021 - June 27, 2021*, Virtual, Nanjing, China, 2021: Institute of Electrical and Electronics Engineers Inc., pp. 165-170.

[21] J. Leng, H. Cai, W. Wang, and Z. Ma, "Double stage Siamese network object tracking algorithm based on template update," in *2021 International Conference on Electronic Information Engineering and Computer Science, EIECS 2021, September 23, 2021 - September 25, 2021*, Changchun, China, 2021: Institute of Electrical and Electronics Engineers Inc., pp. 140-143.

[22] X. Zhou, T. Yin, V. Koltun, and P. Krahenbuhl, "Global Tracking Transformers," in *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, June 19, 2022 - June 24, 2022*, New Orleans, LA, United states, 2022, vol. 2022-June: IEEE Computer Society, pp. 8761-8770.

[23] S. S. Nijhawan, L. Hoshikawa, A. Irie, M. Yoshimura, J. Otsuka, and T. Ohashi, "Efficient Joint Detection and Multiple Object Tracking with Spatially Aware Transformer," arXiv, 2022.

[24] D. Guo, J. Hu, and F. Yan, "YOLOX-EC: A Pedestrian and Vehicle Detection Algorithm in Automatic Driving Scenes," in *6th International Conference on Electronic Information Technology and Computer Engineering, EITCE 2022, October 21, 2022 - October 23, 2022*, Virtual, Online, China, 2022: Association for Computing Machinery, pp. 853-859.

[25] Y. Zhang *et al.*, "ByteTrack: Multi-object Tracking by Associating Every Detection Box," in *17th European Conference on Computer Vision, ECCV 2022, October 23, 2022 - October 27, 2022*, Tel Aviv, Israel, 2022, vol. 13682 LNCS: Springer Science and Business Media Deutschland GmbH, pp. 1-21.

[26] P. Lu, Y. Ding, and C. Wang, "Multi-small target detection and tracking based on improved yolo and sift for drones," *International Journal of Innovative Computing, Information and Control,* vol. 17, no. 1, pp. 205-224, 2021.

[27] J. Luiten *et al.*, "HOTA: A Higher Order Metric for Evaluating Multi-object Tracking," *International Journal of Computer Vision,* vol. 129, no. 2, pp. 548-578, 2021.

[28] Y. Cui, C. Zeng, X. Zhao, Y. Yang, G. Wu, and L. Wang, "SportsMOT: A Large Multi-Object Tracking Dataset in Multiple Sports Scenes," in *2023 IEEE/CVF International Conference on Computer Vision, ICCV 2023, October 2, 2023 - October 6, 2023*, Paris, France, 2023: Institute of Electrical and Electronics Engineers Inc., pp. 9887-9897.