Sensors and Materials, Vol. 36, No. 3 (2023) 875–889
MYU Tokyo

S & M 3569

SeaVit: Sports Event Athletes Tracking Model Design

Linlin Yuan,[1] Yao Liu,[2] and Hsuan-Ming Feng[3*]

[1]College of Physical Education, Huaqiao University, Xiamen 361021, China
[2]College of Medicine, Huaqiao University, Quanzhou, Fujian 362021, China
[3]Department of Computer Science and Information Engineering, National Quemoy University, Kinmen 89250, Taiwan

(Received March 6, 2023; accepted October 17, 2023)

Keywords: sports event, target tracking, transformer, attention module

In recent years, the rapid development of the sports industry has led to increasing attention on complex sports scenes such as basketball and soccer. One of the challenges in these scenes is tracking athletes in motion. To address this issue, we used a large dataset of images captured by a perception visual system as our training data. In this paper, we focus on tracking athletes in complex sports scenes such as basketball and soccer, and propose a tracking model that incorporates a target search attention mechanism and integrates saliency heat maps for feature extraction and target information integration. Our proposed SeaVit method significantly improves tracking performance compared with existing discriminative correlation filter (DCF-based) techniques. Our experimental results demonstrate that our method outperforms existing methods in both precision and real-time performance. With an average distance precision (DP) of 80.5, our proposed tracker outperforms other trackers, achieving the highest DP value. Our study has broad practical applications. In the future, our method can be embedded in visual sensors and applied to athlete training, which helps to improve the effectiveness of athlete training and competition performance.

1. Introduction

With the advancement of sensory vision systems, many known technologies based on the concept of recognizing actions in sports have been developed. For example, the traditional training approach, which relied solely on the coach's intuition, can no longer help enhance the level of competition.(1) The conventional manual assessment lacks a practical basis and is based only on the coach's own experience, which lacks impartiality. Computer vision technology has provided coaches with a new training tool.(2) Because machine vision has more precision and memory than the human eye, it can catch motion targets rapidly and record various motion data of the targets, and these motion data give a more understandable picture of athletes' excellent or poor actions. Because the movement of sports targets in sports videos is quick and frequently varies significantly, generic motion models struggle to forecast the precise location of sports targets. Furthermore, the sports video background is intricate, and the camera moves constantly.

*Corresponding author: e-mail: hmfenghmfeng@gmail.com
https://doi.org/10.18494/SAM4377

ISSN 0914-4935
https://myukk.org/

As a result, to increase the accuracy and robustness, and minimize the complexity of the target tracking algorithm, a new target tracking algorithm for the features of sports videos must be proposed.[3]

In recent years, with the rapid development of the sports industry, the tracking of targets in complex sports scenes, represented by basketball and soccer, has gradually attracted attention. Compared with the most widely studied pedestrian surveillance scenarios, the target tracking problems in sports scenarios are more challenging, such as more severe occlusion and similar appearance interference, more drastic pose changes, and more complex forms of motion.[4] In this paper, we take the tracking of players in complex sports scenes such as basketball and soccer as the research object, and propose a variety of methods for ball and player tracking in complex sports scenes, opening up new avenues for higher semantic level research in sports video analysis, such as action recognition, event detection, and so on, to better identify and correct athlete problems.

One of the core challenges in computer vision is generalized visual object tracking. The job involves evaluating the status of a target item in each frame of a video series while only providing the starting target location. One of the most difficult challenges in object tracking is learning how to recognize target objects in the absence of annotations. Target tracking algorithms are the primary option for this task; however, there are several unresolved issues with present target tracking algorithms, particularly the lack of target tracking accuracy in complicated circumstances. In a typical target tracking situation, a box is used to indicate one or more targets to be monitored in the first frame of the video, and then the defined targets are detected and continuously tracked in consecutive video frames. Among the available approaches, the use of discriminative correlation filters (DCFs)[5–7] has proven to be effective. These approaches learn the target model by minimizing the discriminative target function in each frame to locate the target function. The target model is often configured as a convolution kernel, which gives a compact and generalizable description of the tracked object, contributing to the popularity of DCFs. In DCFs, the objective function combines prior foreground and background knowledge frameworks to deliver successful global inference during learning. However, it induces a significant inductive bias into the anticipated target model. The model predictor has limited flexibility since the objective model is created by minimizing the preceding objectives. In this instance, it is unable to include any previously learnt prior elements into the anticipated target model.

Transformers, on the other hand, have been proven to give powerful global reasoning and to deliver robust global inference in different frameworks because of the utilization of self-attention and cross-attention. As a result, transformers have been successfully deployed in broad object tracking.[8] We offer a novel tracking paradigm in this paper to bridge the gap between DCFs and transformers. To locate the target, our method, such as the use of DCFs, employs a compact target model. However, the weights of this model are determined by a transformer-based model predictor, allowing us to develop a target model more robust than a DCF-based model. This is accomplished by providing a new encoding module for the target states, which allows the transformer to utilize this information more effectively. We next utilize our model predictor to generate weights for a bounded box regressor network so that its predictions match the current

goal. Our proposed SeaVit method exceeds current transformer-based trackers and significantly outperforms the state-of-the-art DCF-based solution in tracking performance.

2. Related Works

In the field of computer vision, target tracking is a challenging problem due to the complex and dynamic nature of the environment. The traditional features are often insufficient to adapt to fast-motion weak target tracking circumstances, while the targets derived using depth features may lack edge information, leading to inaccurate localization and error accumulation. Moreover, the constantly changing target and backdrop make it difficult to gather current global data, resulting in tracking drift. To address these challenges, a robust model for target tracking should focus on two main elements: strong features and effective template updating. In the context of sports events, athlete tracking is of considerable importance for both the sports system and the perception system.[9] In this study, we aim to address the challenges of target tracking in sports events and propose a novel approach to improve the accuracy and robustness of athlete tracking. In this section, we will review the current domestic and international target tracking algorithms, including correlation filtering and deep-learning-based algorithms, and discuss their strengths and limitations in the context of sports events. By combining the review of previous work with the proposed approach, we aim to highlight the significance of our work in advancing the field of athlete tracking.

By reducing the target, DCF-based approaches train a target model to identify the target from the background. Fourier-transform-based solvers have long dominated trackers in DCF-based systems.[6,7,10] Danelljan *et al.*[7] solved the optimization problem using conjugate gradients and a two-layer perceptron as the goal model. Several ways have recently been proposed to enable end-to-end training by converting the tracking issue into a meta learning problem.[11] These methods are based on the concept of untangling the number of iterations of an iterative optimization algorithm and incorporating it into the tracking pipeline to accomplish end-to-end training. Goutam *et al.*[5] used an optimization approach to build a discriminative feature space and forecast the weights of the target model depending on the target state in the initial frame. However, DCF-based approaches cannot be suitable for tracking fast-moving or weak targets. One reason for this is that DCF-based trackers rely on the assumption that the target motion is smooth and predictable, which may not be valid for fast-moving objects. When the target moves rapidly, the model may not be able to adapt quickly enough to keep up with the changes in the appearance and motion of the target. Although recent advances in the field have attempted to address these limitations, such as end-to-end training and discriminative feature space optimization, they cannot fully overcome the challenges posed by fast-moving or weak targets. As such, alternative approaches, such as the use of deep-learning-based trackers, can be more suitable for these scenarios.

The transformer model has recently gained popularity for target tracking owing to its powerful representation learning ability and contextual modeling capacity.[8,12] Unlike recurrent neural networks that model sequence information through recurrent connections across time steps, the transformer employs an attention mechanism to directly model the relationship

between any two elements in the input sequence, which enables it to capture long-range dependences.[13] This property makes the transformer well suited for target tracking, where the model needs to associate the target in the current frame with the corresponding target from potentially many previous frames. The self-attention mechanism also allows the transformer to learn contextual feature representations by aggregating information from the entire input sequence.

Several recent works have employed the transformer for target tracking with encouraging results. TrDiMP[14] encodes target state information through a transformer and predicts target model weights as well as performs bounding box regression. Unlike TrDiMP that uses two separate cross-attention modules in the decoder, our method integrates target state information through two encoding modules before the transformer. Different from the above transformer-based trackers that combine training and testing features in the decoder, STARK[15] stacks and processes them jointly with a single decoder query generating target bounding box predictions. Our method employs the same transformer architecture as DETR,[16] but uses a model predictor instead to predict target classifier weights and bounding box adjuster weights.

In summary, in this paper, we suggest SeaVit based on the transformer model. First, we create a target–search attention (TSA) module to unify feature extraction and target information integration; second, we achieve robust tracking via a TSA-based backbone and a localization head; finally, we create a conditional score-based target template update mechanism to handle target deformation during tracking. The experimental findings reveal that the proposed method tracks well in fast motion scenarios and outperforms the competition in a complete performance comparison.

3. SeaVit Model Generation

In this part, we introduce SeaVit, our end-to-end tracking framework based on the TSA module. First, we present our proposed TSA module, which aims to unify feature extraction and target information integration. This synchronized processing will improve the compatibility of our feature extraction technique with the appropriate tracking objectives. Furthermore, it enables a broad integration of target information, hence better capturing the association between targets and search areas. The full tracking structure is then presented, which consists of merely a TSA-based backbone and a localization head. Finally, we explain SeaVit's training and inference, including the development of a conditional score-based target template updating method to deal with object deformation during tracking.

3.1 Patch embedding

Our model is fed two images as input: a target image $z \in \mathbb{R}^{3\times H_Z \times W_Z}$ and a bigger search image $x \in \mathbb{R}^{3\times H_Z \times W_Z}$. In general, the target object z is at the center, whereas x signifies the broader region in the following frames that contain the target. The two images are input into a convolutional layer ϕ_p^0 with a kernel size of 7 × 7 and a step size of 4, and then into a layer normalization (LN) layer in the patch embedding (Pa.E) stage. We embed the image in the f_z^0

and f_x^0 feature maps as seen below:

$$f_z^0, f_x^0 = LN\left(\phi_p^0(z)\right), LN\left(\phi_p^0(x)\right), \tag{1}$$

where $f_z^0 \in \mathbb{R}^{C_0 \times \frac{H_Z}{4} \times \frac{W_Z}{4}}$, $f_x^0 \in \mathbb{R}^{C_0 \times \frac{H_x}{4} \times \frac{W_x}{4}}$, and C_0 is the number of channels.

3.2 TSA module

Our TSA module receives the target template and the search region as input. It seeks to extract the remote properties of the target template and the search area, respectively, while also fusing the interaction information between them. Unlike the original multi-head attention,[17] the TSA module executes a dual-attention operation on two separate markers for target template and search area sequences. It pays attention to the markers in each sequence to record precise information about the target or search. It performs cross-attention on the tokens in both sequences at the same time in order to communicate between the target template and the search region. A tandem series of tokens can easily execute this hybrid attention system technique. Formally, we separate a tandem token containing multiple targets and searches into two halves before reshaping it into a two-dimensional feature map. A separable deep convolutional projection layer is applied to each feature map to accomplish the extra modeling of local spatial environmental components (i.e., query, key, and value). It also increases efficiency by allowing for downsampling in the key and value matrices. The attended operations for the query, key, and value are then produced by linear projection on the feature maps for each target and search. We use q^t, k^t, and v^t to represent targets and q^s, k^s, and v^s to represent search regions. The TSA is denoted as

$$Att(Q,K,V) = Sm\left(\frac{QW^Q * \left(KW^K\right)^T}{\sqrt{d_k}}\right) * VW^V. \tag{2}$$

3.3 SeaVit for tracking

We created SeaVit, a lightweight end-to-end tracking framework, based on the TSA module. The fundamental idea behind SeaVit is to extract the linked properties of the target template and the search area gradually and to thoroughly integrate data across them. It is made up of two parts: a backbone made up of an iterative target search TSA and a basic localization head that generates the target bounding box. When compared with other popular trackers, separating the stages of feature extraction and information integration results in a more compact and clean tracking pipeline with simply a backbone and a tracking head, no explicit integration modules, and no postprocessing. The entire architecture is shown in Fig. 1.

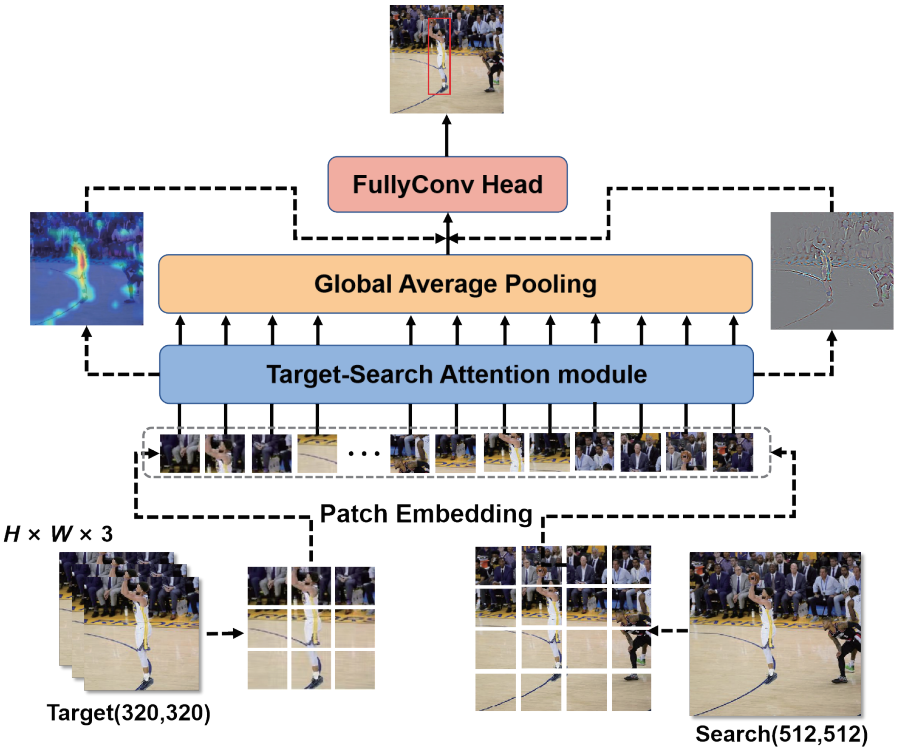

Fig. 1. (Color online) SeaVit for sports event athlete tracking is proposed in this paper, which contains a TSA module, a backbone, and a localization head. SeaVit performs regression using the saliency heat map to obtain bounding boxes.

3.4 Visual explanations

The bias gradient in convolutional networks has an easily observed spatial structure. Consider one convolutional filter, $z = w * x + b$. We can now use the bias gradient to view the mapping of each neuron and layer in such a network. Visualizing the spatial mapping of each convolutional filter yields each neuron mapping. The neuron maps are aggregated to create each layer map. We show these maps for the images after executing normal postprocessing measures to guarantee an optimal viewing contrast. These postprocessing procedures are basic rescaling operations, commonly accompanied by absolute value operations, which are used to show simply the magnitude of importance while ignoring symbols. To avoid disregarding the symbols, it is also feasible to show the positive and negative regions of the map individually. Let us indicate such postprocessing activities by $\psi(\cdot)$. Such postprocessing for a smaller version of the input map also involves scaling operations, which are often performed by conventional techniques. By aggregating such hierarchical mappings, we may also see the estimated network-wide saliency map. By letting c run across channel c_1 at layer l in the neural network, the FullGrad[18] saliency map $S_f(x)$ is given by

$$S_f(x) = \psi\left(\nabla_x f(x) \otimes x\right) + \sum_{l \in L} \sum_{c \in c_l} \psi\left(f^b(x)_c\right). \tag{3}$$

Here, $\psi(\cdot)$ is the postprocessing operator discussed above. In this study, we chose $\psi(\cdot)$ = bilinearUpsample(rescale(abs(·))), where rescale(·) linearly rescales the value to between 0 and 1, and bilinear Upsample(·) uses bilinear interpolation to provide the gradient map the same spatial size as the images.

We emphasize that the FullGrad saliency map described here is approximate in the sense that the full representation is actually $G = \left(\nabla_x f(x), f_b(x)\right) \in \mathbb{R}$, whereas our network-wide saliency map just attempts to combine information from several maps into a visually consistent map. We discovered that aggregating maps according to Eq. (2) gives the clearest maps because it allows neural intelligent maps to vote independently on the value of each geographical position.

3.5 Training and inference

The training process for our tracker essentially follows the usual tracker training protocol.[8,15] We initially train the TSA using the CVT model[19] before adapting the complete tracking architecture to the target dataset. Typically, the L_1 loss and GIoU loss[20] are combined as

$$L_{loc} = \lambda_{L_1} L_1\left(B_i, \hat{B}_i\right) + \lambda_{giou} L_{giou}\left(B_i, B_i\right), \tag{4}$$

where $\lambda_{L_1} = 5$, $\lambda_{L_{giou}} = 2$ is the weight of the two losses, B_i is the ground-truth bounding box, and $\hat{B}_i$ is the predicted bounding box of the targets.

Online templates are useful for acquiring time information as well as dealing with object deformation and appearance changes. Low-quality templates, on the other hand, are usually considered to result in poor tracking performance. As a result, we propose a score prediction module (SPM) to pick dependable online templates on the basis of expected consistency scores. Two attention blocks and a three-level perceptron comprise the SPM. First, a learnable score token functions as a query in the search for ROI tokens. It enables the score token to encode the mined target data. The score token then concentrates on all places of the original target token to compare the mined target with the first target indirectly. Finally, the MLP layer and s-type activation create the score. When the online template's prediction score is less than 0.5, it is deemed negative. For SPM training, which comes after backbone training, we utilize a conventional cross-entropy loss:

$$L_{score} = y_i \log\left(p_i\right) + \left(1 - y_i\right) \log\left(1 - p_i\right), \tag{5}$$

where y_i is the ground-truth label and p_i is the predicted confidence score. Multiple templates, including a static template and N dynamic online templates, as well as cropped search areas, are fed into the mixer throughout the inference phase to build target bounding boxes and consistency

scores. We only update the online templates when the update interval is reached, and the sample with the greatest consistency score is chosen.

4. Tracking Results

4.1 Implementation details

Our tracker was created using Python 3.7 and PyTorch 1.12.1. SeaVit was trained on an RTX2080Ti GPU. SeaVit, in particular, is a useful tracker that does not need postprocessing, positional embedding, or a multilayer feature aggregation architecture. SeaVit's backbone was pretrained on ImageNet[11] using CVT-21[19] (the first 16 layers were employed). The training set is made up of Multi-sports datasets.[12] SeaVit's training approach is divided into two stages. It comprises the first 300 epochs for the backbone and head, as well as extra 60 epochs for the score prediction head. SeaVit is trained using ADAM[21] with a weight decay of 10^{-4}. Over 200 calendar hours, the learning rate declines from $1 \times e^{-4}$ to $1 \times e^{-5}$. The sizes of the search image and template are 512 × 512 and 320 × 320 pixels, respectively. As input to SeaVit, we utilize the initial template, other internet templates, and the current search region. When an update interval of 200 was achieved, dynamic templates were updated. To replace the previous template, the template with the greatest prediction score in that period is chosen.

4.2 Datasets

Using Multi-sports datasets, we validate the benchmark performance of our proposed SeaVit. Li *et al.* proposed Multi-sports,[12] a new spatio-temporal local motion multiplayer dataset, by first analyzing the key components for building a realistic and challenging dataset for spatio-temporal motion detection by proposing three criteria: (1) multiplayer scenes and motion-related identification, (2) well-defined boundaries, and (3) relatively fine-grained highly complex granularity classes. The Multi-sports v1.0 datasets are built using these principles by selecting four sports categories and gathering 3200 video clips with 902k bounding boxes annotated with 37701 action occurrences. The Multi-sports dataset is notable for its vast diversity, extensive annotation, and excellent quality.

4.3 Visualization of attention maps

Figure 2 depicts a range of attention maps used to investigate how blended attention operates in the blender backbone. We deduced the following from the four forms of attention maps: (i) interference with the backdrop inhibits the layer, (ii) online templates may be more adaptable to changes in appearance and aid in target differentiation, and (iii) the foregrounds of many templates can improve mutual cross-attention. Figure 2 depicts our findings. We use the heat maps of the trained model on Multi-sports for various visual explanations. It is worth noting that all of them, including gradcam++,[22] ablationcam,[23] and Fullgrad,[18] may focus on the tracking target region. As seen in the illustration, Fullgrad can more precisely find items. In contrast, the

Fig. 2. (Color online) Our proposed SeaVit uses FullGrad to obtain the best region of interest in different saliency heat maps.

eigencam[24] model only considers the immediate surroundings. Our SeaVit, on the other hand, provides fine-grained supervision for tokens from diverse areas, which might assist the model to better pinpoint the tracking target.

4.4 Experiment and performance comparison results

To validate the performance of our proposed tracker, we compare it with 10 state-of-the-art discriminative correlation filter-based trackers, namely, SRDCF,[25] LMCF,[26] LCT,[27] KCF, [28] CSK,[29] SAMF,[30] DSST,[31] DCF CA,[32] SAMF CA,[32] and MOSSE CA.[32] All of the videos in the datasets are annotated with four qualities that comprise several hard settings, such as basketball, soccer, volleyball, and gymnastics. As illustrated in Fig. 3, we give the outcomes of partial attributes. The distance precision (DP) is defined as the proportion of frames whose projected position is within a particular threshold distance of the ground truth, which is typically

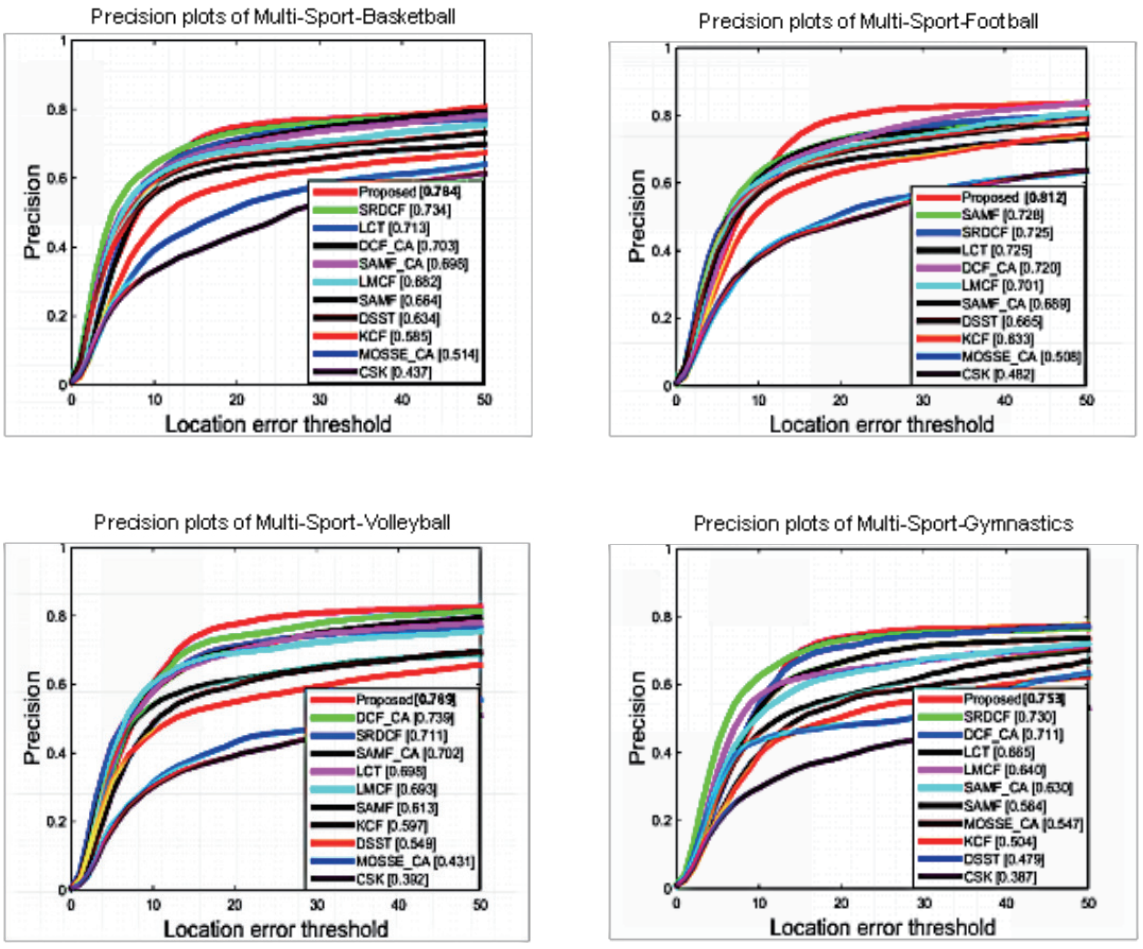

Fig. 3. (Color online) Attribute-based evaluation of distance precision and overlap success plots over four challenges of basketball, soccer, volleyball, and gymnastics. The number of sequences for each attribute is shown in brackets.

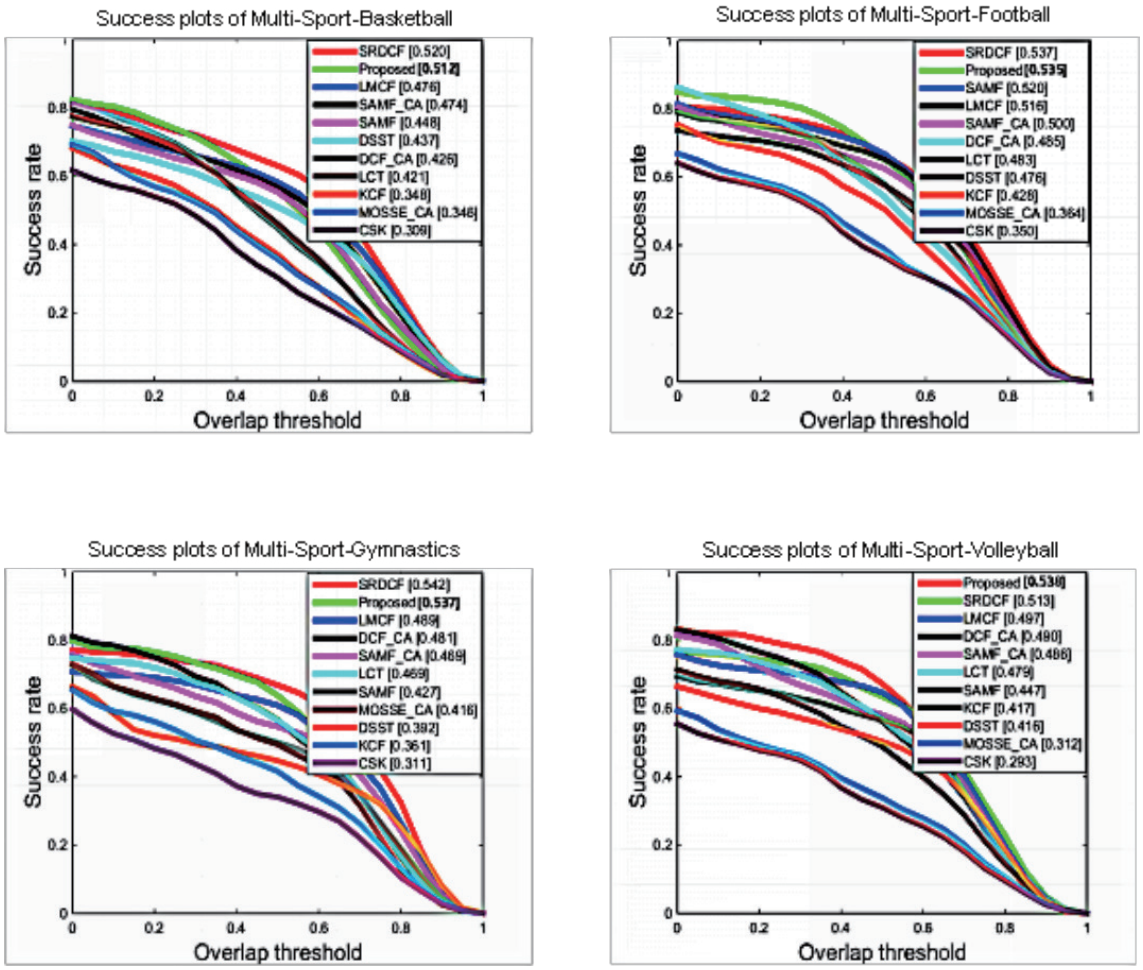

Fig. 3. (Color online) (Continued) Attribute-based evaluation of distance precision and overlap success plots over four challenges of basketball, soccer, volleyball, and gymnastics. The number of sequences for each attribute is shown in brackets.

set at 20 pixels. The overlap successful plot (OP) is defined as the proportion of frames with an overlap rate greater than a given threshold, which is often set at 0.5. We can see that our tracker outperformed the competition in virtually all of the reported metrics.

Figure 3 indicates that the proposed method performs well with DP and OPs in four attribute challenges, achieving outstanding DP in basketball (78.4%), soccer (81.2%), volleyball (78.9%), and gymnastics (75.3%). The overlap success plots perform second best in all of the exhibited criteria. These findings show that an efficient model update strategy can increase tracking accuracy, particularly in soccer attributes.

Table 1 shows the distance accuracy and overlap success plots of eleven trackers using Multi-sports datasets. As shown in Table 1, the proposed tracker outperforms existing trackers in DP and overlap success (OS). With an average DP of 80.5, the suggested tracker outperforms SRDCF (75.4), LMCF (72.8), and LCT (76.7). The OS plots retain similar accuracy (0.538) to

Table 1
Tracking results of DP and OS comparisons with state-of-the-art trackers performed on Multi-sports datasets.

	DP (%)	OS (%)	Speed (fps)
SRDCF	75.4	55.2	5.3
LMCF	72.8	53.0	81.6
LCT	76.7	50.2	20.5
SAMF	68.3	48.7	17.8
DSST	65.2	47.4	27.6
KCF	63.4	42.1	210.4
CSK	47.8	35.0	264.3
SAMF_CA	72.0	51.2	12.72
DCF_CA	74.0	49.3	90.9
MOSSE_CA	54.0	38.6	122.6
Proposed	80.5	53.8	128.2

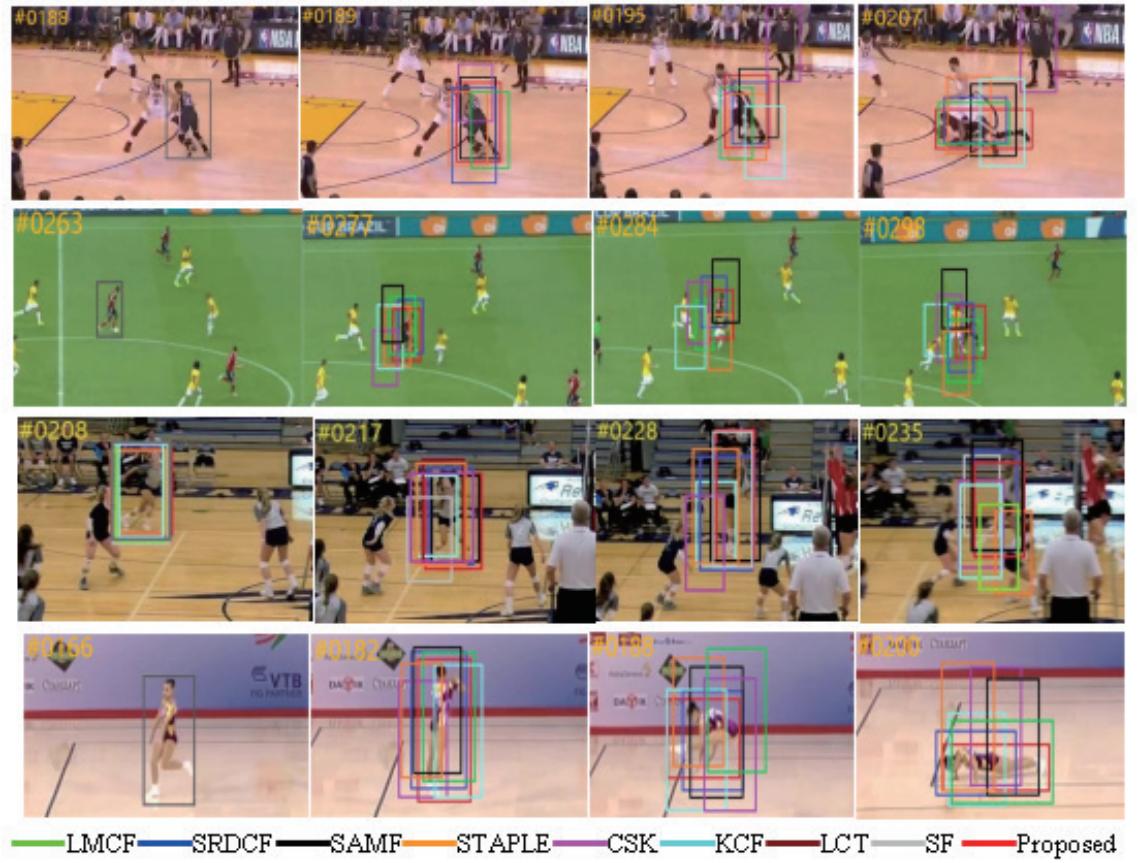

Fig. 4. (Color online) Qualitative comparison of proposed approach with eight trackers on four challenge sequences (from left to right and top to down are basketball, soccer, volleyball, and gymnastics). SF represents the position of the starting frame, after which the tracking starts.

SRDCF (0.552) and exceed LMCF (0.530). In terms of speed, our tracker relies heavily on the computing efficiency of CFs in the frequency domain and a variety of hand-crafted characteristics for tracking. The tracking speed of our tracker beats those of the SRDCF (5.3),

LCT (20.5), and DSST (27.6) trackers, which achieved a real-time speed of 128.2 frames per second. These findings indicate the capacity of the tracking model to update adaptively and integrate efficiently various characteristics for feature representation.

Figure 4 shows qualitative comparisons of the proposed tracker with seven other trackers on Multi-sports datasets, namely, LMCF, SRDCF, SAMF, STAPLE, CSK, KCF, and LCT. We can show that the suggested tracker performs effectively in occlusion settings (basketball), fast motion scenarios (soccer and volleyball), and motion blur scenarios (gymnastics)

In the basketball sequence, we know that the target faces an occlusion problem from #0189 to #0195; nonetheless, our tracker can still reliably detect the target item. In the event of a severe occlusion, we chose not to update the tracking model. Other trackers, such as KCF, cause tracking drift after severe blockage, demonstrating the efficacy of the presented approaches.

In the soccer and volleyball sequences, the target item faces problems such as quick motion. The object is not tracked by the KCF, CSK, LCT, and LMCF trackers. Because we incorporate numerous robust discriminative features and an updating method in our tracker, we can effectively follow the object from the beginning to the end of sequences.

In the gymnastics sequence, the target mainly faces challenges such as motion blur. Most trackers, such as LMCF and KCF, lose the object and produce drift; our tracker can track the object accurately; this illustrates that the suggested approach is resistant to motion blur.

5. Conclusions

SeaVit is a vision transformer-based tracking framework in a real-life sensory scene for athletes in sporting events, which aims to integrate feature extraction and target integration, resulting in a tidy and compact tracking pipeline. The TSA module extracts features while also interacting with target templates and search areas. We conducted empirical studies to evaluate the performance of our proposed method. The experimental results show that SeaVit outperforms other popular trackers significantly in terms of short-term tracking. Specifically, our proposed method achieves higher accuracy and real-time performance than the state-of-the-art DCF-based technique. With an average DP of 80.5, the suggested tracker outperforms SRDCF (75.4), LMCF (72.8), and LCT (76.7). The overlap success plots retain similar accuracy (0.538) to SRDCF (0.552) and exceed LMCF (0.530). In terms of speed, our tracker relies heavily on the computing efficiency of CFs in the frequency domain and a variety of hand-crafted characteristics for tracking.

Our proposed method has potential applications in various fields, especially in sports and fitness. For example, our method can be used to track the movements of athletes during training sessions, providing valuable insights for coaches and trainers to improve their performance. Moreover, our method can also be used to monitor the movements of patients during rehabilitation, allowing healthcare professionals to provide personalized treatment plans on the basis of the patient's progress.

In addition, our proposed method can be integrated with various sensors, such as cameras and wearable devices, to ensure the real-time tracking and monitoring of the target object. This can be particularly useful in sports and fitness applications, where accurate and real-time

tracking is essential for performance analysis and injury prevention. In future work, we plan to further explore the potential applications of SeaVit in various fields and investigate the integration of different sensors to improve the accuracy and robustness of our proposed method.

Acknowledgments

This work was supported by Fujian Provincial Science and Technology Major Project (No. 2020HZ02014) and QuanZhou Science and Technology Major Project (No. 2021GZ1).

References

1 C. Xu and Y. Li: Innovative Comput. **791** (2022) 1799. https://doi.org/10.1007/978-981-16-4258-6_234
2 Z. Richards and P. Gaynor: 2022 IEEE SoutheastCon. (IEEE, 2022) 221. https://doi.org/10.1109/SoutheastCon48659.2022.9764130
3 X. Xuan and H. Xu: Math. Probl. Eng. **2022** (2022). https://doi.org/10.1155/2022/8445250
4 C. Hsia, C. Chien, H. Hsu, J. Chiang, and H. Tseng: J. Intelligent Fuzzy Syst. **36** (2019) 1171. https://doi.org/10.3233/JIFS-169891
5 G. Bhat, M. Danelljan, L. V. Gool, and R. Timofte: Proc. 2019 IEEE/CVF Int. Conf. Computer Vision (ICCV, 2019) 6182.
6 D. S. Bolme, J. R. Beveridge, B. A. Draper, and Y. M. Lui: Proc. 2010 IEEE Computer Society Conf. Computer Vision and Pattern Recognition (CVPR, 2010) 2544–2550. https://doi.org/10.1109/CVPR.2010.5539960
7 M. Danelljan, G. Bhat, F. S. Khan, and M. Felsberg: Proc. 2019 IEEE Computer Society Conf. Computer Vision and Pattern Recognition (CVPR, 2019) 4660–4669.
8 X. Chen, B. Yan, J. Zhu, D. Wang, X. Yang, and H. Lu: Proc. 2021 IEEE Computer Society Conf. Computer Vision and Pattern Recognition (CVPR, 2021) 8126–8135. https://doi.org/10.48550/arXiv.2103.15436
9 C. Hsia, S. Yen, and J. Jang: Sens. Mater. **31** (2019) 1803. https://doi.org/10.18494/SAM.2012.764
10 Y. l. Qian, S. Yan, A. Lukežič, M. Kristan, J. Kämäräinen, and J. Matas: Proc. 2020 IEEE Int. Conf. Pattern Recognition (ICPR, 2020) 7825-7832. https://doi.org/10.1109/ICPR48806.2021.9412984J
11 G. Wang, C. Luo, X. Sun, Z. Xiong, and W. Zeng: Proc. 2020 IEEE Computer Society Conf. Computer Vision and Pattern Recognition (CVPR, 2020) 6288–6297. https://doi.org/10.48550/arXiv.2004.00830
12 Y. Li, L. Chen, R. He, Z. Wang, G. Wu, and L. Wangin: Proc. 2021 IEEE Int. Conf. Computer Vision (ICCV, 2021) 13536–13545. https://doi.org/10.48550/arXiv.2105.07404
13 B. Yu, M. Tang, L. Zheng, G. Zhu, J. Wang, H. Feng, and H. Lu: Proc. 2021 IEEE/CVF Int. Conf. Computer Vision (ICCV, 2021) 9856–9865.
14 N. Wang, W. Zhou, J. Wang, and H. Li: Proc. 2021 IEEE Computer Society Conf. Computer Vision and Pattern Recognition (CVPR, 2021) 1571–1580. https://doi.org/10.48550/arXiv.2103.11681
15 B. Yan, H. Peng, J. Fu, D. Wang, and H. Lu: Proc. 2021 IEEE/CVF Int. Conf. Computer Vision (ICCV, 2021) 10448–10457. https://doi.org/10.48550/arXiv.2103.17154
16 N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko: Proc. 2020 European Conf. Computer Vision (ECCV, 2020) 213–229. https://doi.org/10.1007/978-3-030-58452-8_13
17 A. Vaswani, N. Shazeer, N.Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, I. Polosukhin: Proc. Advances in Neural Information Processing Systems (2019) 30. https://doi.org/10.48550/arXiv.1906.05909
18 S. Srinivas and F. Fleuret: Proc. Advances in Neural Information Processing Systems (2019) 32. https://doi.org/10.48550/arXiv.1905.00780
19 H. Wu, B. Xiao, N. Codella, M. Liu, X. Dai, L. Yuan and, L. Zhang: Proc. 2021 IEEE/CVF Int. Conf. Computer Vision (ICCV, 2021) 22–31. https://doi.org/10.48550/arXiv.2111.11432
20 H. Rezatofighi, N. Tsoi, J. Gwak, A. Sadeghian, I. Reid, S. Savarese: Proc. 2019 IEEE Computer Society Conf. Computer Vision and Pattern Recognition (CVPR, 2019) 658–666. https://doi.org/10.48550/arXiv.1902.09630
21 B. Yu, M. Tang, L. Zheng, G. Zhu, J. Wang, H. Feng, H. Lu: Proc. 2021 IEEE/CVF Int. Conf. Computer Vision (ICCV, 2021) 9856–9865. https://doi.org/10.1109/ICCV48922.2021.00971
22 A. Chattopadhay, A. Sarkar, P. Howlader, V. N. Balasubramanian: Proc. 2018 IEEE Winter Conf. Applications of Computer Vision (WACV, 2018) 839–847. https://doi.org/10.1109/WACV.2018.00097
23 S. Desai and H. G. Ramaswamy: Proc. 2020 IEEE Winter Conf. Applications of Computer Vision (WACV, 2020) 983–991. https://doi.org/10.1109/WACV45572.2020.9093360

24 M. B. Muhammad and M. Yeasin: Proc. 2020 Int. Joint Conf. Neural Networks (IJCNN, 2020) 1–7. https://doi.org/10.1109/IJCNN48605.2020.9206626
25 M. Danelljan, G. Hager, and F. S. Khan: Proc. 2015 IEEE/CVF Int. Conf. Computer Vision (ICCV, 2015) 4310. https://doi.org/10.1007/978-3-319-19665-7_10
26 M. Wang, Y. Liu, and Z. Huang: Proc. 2017 IEEE Computer Society Conf. Computer Vision and Pattern Recognition (CVPR, 2017) 4800. https://doi.org/10.48550/arXiv.1703.05020
27 C. Ma, X. Yang, C. Zhang, and M.-H. Yang: Proc. 2015 IEEE Computer Society Conf. Computer Vision and Pattern Recognition (CVPR, 2015) 5388. https://doi.org/10.1109/CVPR.2015.7299177
28 F. Henriques, R. Caseiro, P. Martins, and J. Batista: Proc. 2014 IEEE Trans. Pattern Analysis and Machine Intelligence **37** (PAMI, 2014) 583–596. https://doi.org/10.1109/TPAMI.2014.2345390
29 J. F. Henriques, R. Caseiro, P. Martins, and J. Batista: Proc. 2012 European Conf. Computer Vision (ECCV, 2012) 702. https://doi.org/10.1007/978-3-642-33765-9_50
30 Y. Li and J. Zhu: Proc. 2014 European Conf. Computer Vision (ECCV, 2014) 254. https://doi.org/10.1007/978-3-319-16181-5_18
31 M. Danelljan, G. Hager, F. Khan, and M. Felsberg: Proc. 2014 British Machine Vision Conf. (2014). http://doi.org/10.5244/C.28.65
32 M. Mueller, N. Smith, and B. Ghanem: Proc. 2017 IEEE Computer Society Conf. Computer Vision and Pattern Recognition (CVPR, 2017) 1396. https://doi.org/10.1109/CVPR.2017.152

About the Authors

Linlin Yuan was born in Anhui, China, in 1983. He received his M.S. degree in physical education from Tianjin University of Sport, Tianjin, China, in 2011. In 2016, he was selected by the Federation of University Sports of China to participate in the training of track and field coaches at Arizona State University, U.S.A. He is currently an instructor in the College of Physical Education, Huaqiao University. His current research interests are in the areas of sports training, sports education, and public sports management. (yuanlinlin83@gmail.com)

Yao Liu was born in Fujian, China in 1999. He received his M.S. degree in engineering from National Quemoy University, Taiwan. He is currently with the College of Medicine, Huaqiao University, China. His current research interests are in the areas of deep learning and computer vision. (yowk0529@gmail.com)

Hsuan-Ming Feng received his B.S. degree in automatic control engineering from Feng-Chia University, Taichung, Taiwan, R.O.C., in 1992. He received his M.S. and Ph.D. degrees in computer science and information engineering from Tamkang University, Tamsui, New Taipei City, Taiwan, R.O.C., in 1994 and 2000, respectively. He is currently a full professor with the Department of Computer Science and Information Engineering, National Quemoy University. His current research interests include fuzzy systems, convolutional neural networks, wireless networks, optimal learning algorithms, image processing, and robot system. (hmfenghmfeng@gmail.com)