SCIENCE CHINA
Information Sciences

• RESEARCH PAPER •

February 2026, Vol. 69, Iss. 2, 122107:1–122107:15
https://doi.org/10.1007/s11432-024-4501-3

You can only watch the past: track attention network for online spatio-temporal action detection

Shaowen SU*, Minggang GAN & Yan ZHANG

State Key Laboratory of Intelligent Control and Decision of Complex Systems, School of Automation, Beijing Institute of Technology, Beijing 100081, China

Received 8 February 2024/Revised 14 March 2025/Accepted 16 June 2025/Published online 12 January 2026

Abstract Online spatio-temporal action detection (OSTAD) aims to identify and localize action instances in real-time video streams without accessing future frames. However, the online setting imposes strict constraints of incremental inference, limited memory, and causal processing, which severely restrict the availability of effective information. To address this, we propose the track attention network (TAN), introducing a history-aware track-and-detect paradigm. Instead of detecting actions independently at each frame, TAN leverages historical detection results and spatio-temporal continuity to enhance current-frame features. Specifically, we propose three strategies. First, a history-aware actor distribution prediction strategy estimates current actor distributions based on spatial continuity and appearance similarity. Second, an actor distribution inference strategy via track attention introduces two attention modules—track channel attention and track efficient attention—to model semantic relations among actor distributions for robust fusion. Third, a history-aware feature modulation strategy injects localization priors from actor distributions into action features, improving representation quality and detection accuracy. Extensive experiments on the JHMDB21 and UCF24 benchmarks demonstrate the effectiveness of our method. TAN achieves 80.3% frame-level mAP (f-mAP) and 88.3% video-level mAP (v-mAP) on JHMDB21, and 88.1% f-mAP and 54.8% v-mAP on UCF24, outperforming existing online methods and even several offline approaches.

Keywords online spatio-temporal action detection, track-and-detect, historical detection, actor distribution, track attention

Citation Su S W, Gan M G, Zhang Y. You can only watch the past: track attention network for online spatio-temporal action detection. Sci China Inf Sci, 2026, 69(2): 122107, https://doi.org/10.1007/s11432-024-4501-3

1 Introduction

Online spatio-temporal action detection (OSTAD) [1–4] aims to detect action instances by locating actors and classifying their actions in real-time video streams without accessing future frames. Due to its online operational nature, OSTAD enjoys broader applicability in diverse real-world scenarios, including intelligent surveillance, autonomous driving, and sports entertainment. However, unlike traditional spatio-temporal action detection (STAD) [3, 5–7], OSTAD inherently operates under strict constraints, such as incremental detection, limited memory storage, and stringent causal restrictions. These constraints significantly restrict the quantity and quality of available contextual information, posing substantial challenges compared to STAD. Consequently, methods developed for STAD tasks, which rely on rich contextual information, are difficult to directly transfer to OSTAD. As a result, the development of OSTAD methods remains limited, with relatively fewer advances reported to date.

Existing OSTAD methods predominantly adopt a detect-and-link paradigm, independently detecting actors at each frame and subsequently associating them temporally to form action tubes. However, these approaches [8–11] often perform detection at each moment in complete isolation, neglecting valuable prior information embedded in historical frames and failing to fully exploit the spatio-temporal continuity of actors' movements. Although recent studies [4, 12–15] attempt to leverage short-term temporal context, they still lack the capacity to utilize historical detection data explicitly, limiting their potential for enhancing detection robustness. These limitations highlight the core challenge of OSTAD, the insufficiency of effective information under online constraints, which directly hinders accurate and robust action detection.

To address the challenge of insufficient effective information under online constraints, this paper proposes a history-aware track-and-detect paradigm tailored for OSTAD. Unlike conventional detect-and-link methods, the proposed track attention network (TAN) explicitly leverages historical detections and spatio-temporal continuity to improve detection accuracy. First, to address the lack of continuity in per-frame detection and the intermittent

* Corresponding author (email: su_shaowen@foxmail.com)

absence of appearance cues, we propose a history-aware actor distribution prediction strategy that dynamically associates historical detections based on spatial relations and available appearance features. Second, to mitigate the lack of interaction among predicted actor regions, we design an actor distribution inference strategy via track attention, employing two dedicated attention modules to model semantic relationships under dynamic and sparse conditions. Third, to bridge the modality gap between actor distributions and action features, we develop a history-aware feature modulation strategy that injects spatial priors into the feature representation based on semantic alignment. Extensive experiments on OSTAD benchmarks demonstrate that these contributions collectively lead to state-of-the-art performance, surpassing existing online methods and even outperforming many offline approaches that rely on future or buffered frames.

The remainder of this paper is structured as follows: Section 2 reviews related work; Section 3 elaborates the proposed method in detail; Section 4 provides comprehensive experimental evaluations and analyses; and Section 5 concludes the paper. The implementation code will be released on GitHub after publication (http://github.com/Riiick-2011/TAN).

2 Related work

2.1 Online spatio-temporal action detection

OSTAD methods [9, 10] aim to provide accurate detection results at each newly arrived video frame in an online video stream. OSTAD methods can be broadly categorized based on the temporal extent of information utilized during detection.

(1) Current frame only. Early studies, such as Gkioxari et al. [8], leveraged dual-stream networks integrating appearance information and optical flow for action proposals and classification. Weinzaepfel et al. [9] employed edge-based proposals with frame-wise classification, subsequently linking detections temporally.

(2) Current and historical frames. Methods explicitly using historical frames include Saha et al. [10], who merged region proposals from dual-stream branches based on spatial overlaps. Singh et al. [11] progressively constructed action tubes online by combining RGB and optical flow inputs. Sun et al. [12] and Zhang et al. [14] utilized historical contextual and motion information to improve detection. Recent methods like YOWO [13], YOWOv2 [15], and YWOM [4] further integrated spatio-temporal historical information within unified frameworks.

Existing OSTAD approaches predominantly follow a detect-and-link paradigm, wherein detection results at each frame are completely independent of historical detections. Although some methods incorporate historical frame information, they still fail to utilize historical detection data explicitly, resulting in underutilization of spatio-temporal continuity and limiting the accuracy of detection. In contrast, our proposed track-and-detect framework leverages a history-aware actor distribution prediction strategy to explicitly use historical detection data for the first time. This approach significantly improves both the accuracy and consistency of detection under online constraints.

2.2 Multiple object tracking

Multi-object tracking (MOT) [16,17] aims to detect and track multiple objects across video frames, forming coherent trajectories for each detected object. Conventional MOT methods typically follow a detect-and-track paradigm: first detecting objects independently in each video frame, then associating these detections temporally into tracks using metrics such as intersection over union (IoU) and appearance-based feature similarity. Common algorithms, including the Hungarian algorithm, are employed to solve the data association problem.

Recent MOT advancements incorporate sophisticated methods for better handling dynamic scenarios and occlusions. Gao et al. [18] proposed graph neural networks (GNNs) for dynamic modeling of target relationships. Wan et al. [19] and Zhu et al. [20] employed attention mechanisms and transformers to refine tracking predictions, while Chen et al. [21] utilized motion models for robust bounding box prediction.

Unlike MOT methods, which prioritize tracking existing detections, our work introduces a novel track-and-detect paradigm. By leveraging historical detection results to inform current frame predictions, our method reverses the conventional detect-and-track paradigm and explicitly exploits spatio-temporal continuity, simultaneously improving detection accuracy and tracking robustness.

2.3 Visual attention

Visual attention mechanisms [22] dynamically enhance deep learning models by selectively emphasizing relevant features, effectively handling noisy backgrounds and irrelevant information.

Attention mechanisms can be classified into channel attention and spatial attention. Channel attention methods like SE-Net [23] and ECA-Net [24] emphasize significant channels, while spatial attention methods such as CBAM [25] and SPAN [26] focus on spatially relevant regions.

Attention can also be categorized into self-attention and cross-attention. Self-attention, as used in transformers [27], captures internal semantic relationships within feature sets, whereas cross-attention [28] fuses information across different feature modalities.

In addition to the visual attention mechanisms discussed above, recent progress in adjacent areas, including attention-driven object detection [29, 30], adversarially robust tracking [31], and feature selection for multi-label recognition [32], offers complementary insights that may inspire future enhancements to attention design in video understanding.

Our method introduces an actor distribution inference strategy via track attention. Specifically, we design two specialized attention modules—track channel attention (TCA) and track efficient attention (TEA)—to model semantic relationships among actor distributions. These modules enhance semantic interaction and improve the consistency and reliability of localization predictions.

In addition, we propose a history-aware feature modulation strategy that dynamically adjusts visual feature representations by incorporating prior knowledge derived from actor distributions. This strategy bridges the modality gap between actor distributions and action features, improving the quality of feature representation and enhancing overall detection performance.

3 Methodology

To address the challenge of insufficient effective information in the online setting, we propose a novel TAN that introduces a history-aware track-and-detect paradigm. Unlike conventional detect-and-link approaches, TAN fully leverages historical detection results and spatio-temporal continuity to dynamically enhance feature representation and improve the accuracy of current detection.

As illustrated in Figure 1, the proposed TAN consists of three consecutive stages: feature extraction, tracking-guided representation refinement, and detection memory update.

The feature extraction stage adopts a dual-branch backbone to extract multi-scale features from the input video segment, where the last frame corresponds to the current detection moment. The 2D branch captures spatial cues from the current frame to support precise localization, while the 3D branch encodes spatio-temporal dynamics for action classification. Inspired by YOWO [13], we adopt the channel fusion and attention module (CFAM) to integrate both branches. After concatenating 2D and 3D features along the channel dimension, CFAM computes self-attention across channels to capture dependencies and adaptively fuse complementary spatial and temporal information.

The tracking-guided representation refinement stage, which constitutes the core of our innovation, performs history-aware refinement of action features through three specialized strategies.

(1) History-aware actor distribution prediction strategy. To mitigate the uncertainty and intermittent unavailability of appearance features in historical data, this strategy dynamically associates historical detections based on spatial continuity and appearance similarity, and exploits spatio-temporal continuity to estimate the current actor distribution.

(2) Actor distribution inference strategy via track attention. To address the lack of interaction among predicted actor distributions and the incompatibility of conventional attention mechanisms with distributional properties, this strategy introduces two specialized track attention modules—TCA and TEA—to infer semantic relations among actor distributions, enhancing fusion reliability and consistency.

(3) History-aware feature modulation strategy. To bridge the structural gap between actor distributions and action features, this strategy injects localization-relevant spatial priors from actor distributions into action features based on their semantic relevance, thereby improving the precision of action representation.

The detection memory update stage receives the refined multi-scale features and generates current-frame predictions via multiple detection heads. Results are then filtered using soft-NMS and confidence thresholds before being stored in a historical detection buffer. This memory is maintained in a sliding window fashion to support efficient prediction and tracking across time without excessive storage.

3.1 History-aware actor distribution prediction strategy

The motion of action actors follows physical laws, exhibiting continuity in spatial position, shape, and appearance. However, historical detections often fail to align perfectly with ground truth due to inherent uncertainty and

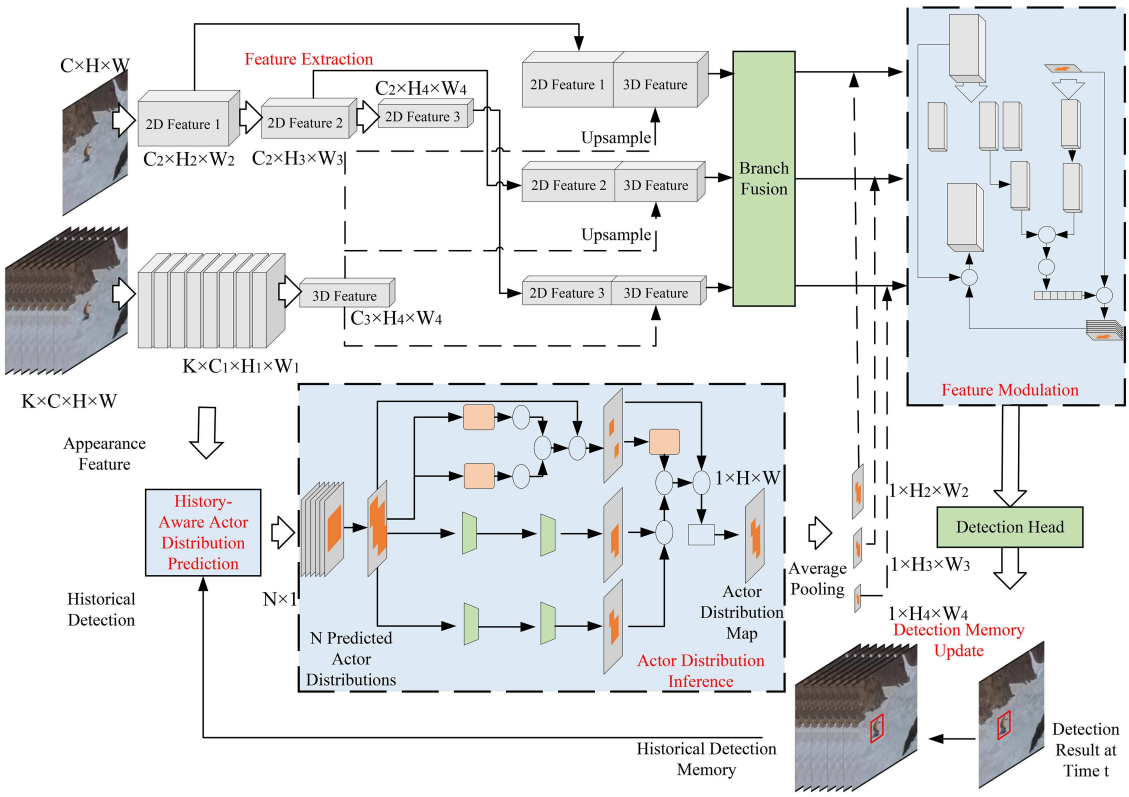

Figure 1 (Color online) Overview of the track attention network (TAN). The network comprises three stages: feature extraction, tracking-guided representation refinement, and detection memory update. In the first stage, three different scales of spatial features are extracted from the current frame via a 2D CNN branch, while a single-scale spatio-temporal feature is extracted from a segment consisting of both the current and historical frames via a 3D CNN branch. These two types of features are then fused along the channel dimension through branch fusion to produce three features at different scales, which are subsequently used for feature modulation in the second stage. The second stage, which forms the core of our contribution, takes as input the feature sequence from the 3D branch together with the historical detection results, and incorporates three proposed strategies for history-aware refinement of action features (see Subsections 3.1–3.3). In the third stage, the three features at different scales are respectively fed into three parallel detection heads to generate current-frame detection results, and the historical detection memory is updated in a sliding window manner to support continual detection refinement over time.

incomplete appearance information. Specifically, historical detections may not fully cover the actor regions, and not all historical frames are included in the current video segment input, resulting in the absence of corresponding visual features for some detections.

To address these challenges, we propose a history-aware actor distribution prediction strategy. This strategy dynamically links historical detections into actor trajectories using spatial overlap and appearance similarity, and then predicts the actor distribution at the current time step based on spatio-temporal continuity. An overview of the linking process is illustrated in Figure 2.

3.1.1 *Trajectory initialization and update*

Each active trajectory maintains the motion and appearance states of an actor. For each unmatched historical detection, a new trajectory is initialized using its bounding box observation $[x, y, a, h]$, where x and y are the center coordinates, h is the height, and a is the aspect ratio. The state vector $\boldsymbol{S} = [x, y, a, h, \dot{x}, \dot{y}, \dot{a}, \dot{h}]$ is estimated using a Kalman filter, where the velocity terms are initialized as zero. If the detection has a corresponding visual feature $\boldsymbol{f}^{\mathrm{B}}$, the trajectory appearance $\boldsymbol{f}^{\mathrm{U}}$ is initialized as

$$\boldsymbol{f}^{\mathrm{U}} = \frac{\boldsymbol{f}^{\mathrm{B}}}{\|\boldsymbol{f}^{\mathrm{B}}\|}. \tag{1}$$

During updates, matched detections refine the trajectory's state through Kalman filter prediction and correction. If appearance features exist, we apply exponential moving average (EMA) to update the appearance as

$$\boldsymbol{f}^{\mathrm{U}} = \alpha \cdot \boldsymbol{f}^{\mathrm{U}} + (1-\alpha) \cdot \frac{\boldsymbol{f}^{\mathrm{B}}}{\|\boldsymbol{f}^{\mathrm{B}}\|}, \quad \alpha = 0.9. \tag{2}$$

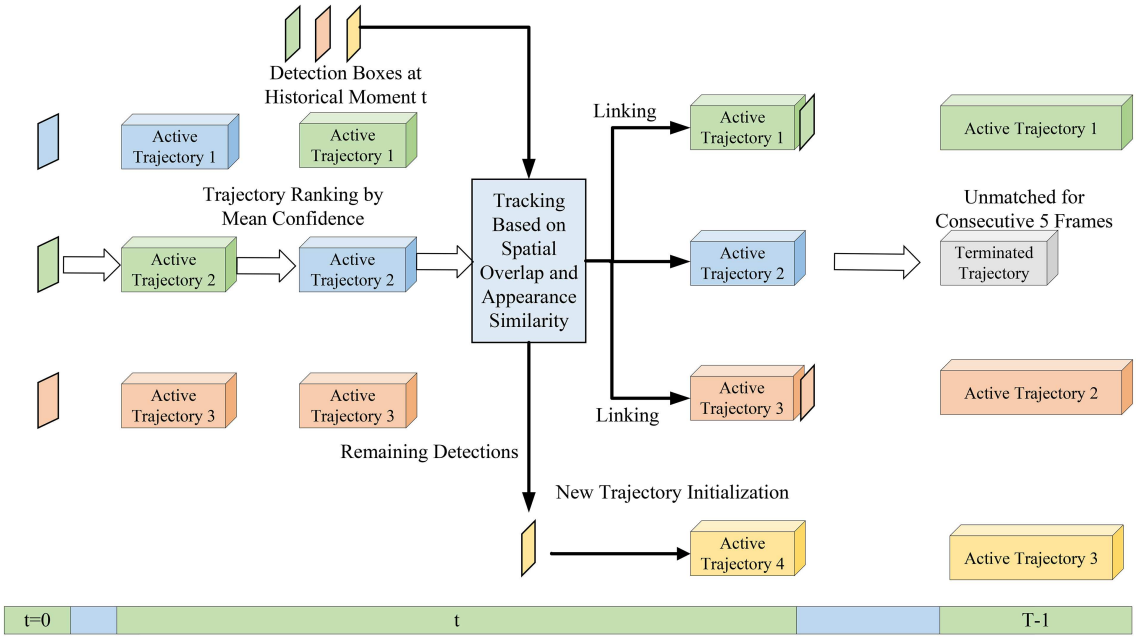

Figure 2 (Color online) Illustration of the linking process at moment t. Active trajectories are sorted by score and linked with detections based on spatial overlap and appearance similarity. Unlinked detections are used to initialize new trajectories, and unmatched trajectories are marked inactive after a timeout.

3.1.2 *Trajectory linking*

At each historical time step t, active trajectories are ranked by their average detection confidence. Each trajectory is linked to the best candidate detection based on spatial overlap (IoU) and appearance similarity:

$$s_{\text{as}} = 1 - \frac{1}{1 + \exp\left(\frac{\boldsymbol{f}^{\text{U}} \cdot \boldsymbol{f}^{\text{B}}}{\|\boldsymbol{f}^{\text{U}}\| \|\boldsymbol{f}^{\text{B}}\|} - 1\right)}. \tag{3}$$

If a trajectory fails to match any candidate detection with sufficient overlap, it is marked as missed. After a fixed number of consecutive misses, it is deactivated.

3.1.3 *Distribution prediction*

At the final historical time step, surviving trajectories use their Kalman filter states to predict the spatial distribution of actors at the current time step T. These predicted distributions serve as priors for subsequent feature interaction and detection.

Each actor distribution can be intuitively viewed as a soft spatial extension of its corresponding bounding box, where the confidence value peaks at the center and gradually decays toward the boundaries. Unlike hard bounding boxes, this spatially diffuse representation allows the model to express uncertainty and capture region-level importance, making it more robust for downstream tasks like attention-based feature refinement.

This strategy enables robust actor distribution prediction by leveraging incomplete and noisy historical data under online constraints.

3.2 Actor distribution inference strategy via track attention

To improve the reliability of coarse actor distributions predicted from historical trajectories, we propose an inference strategy that leverages track attention to model contextual dependencies among predicted actors. Specifically, we aim to capture semantic relationships, such as redundancy, complementarity, and alignment, between distributions by embedding them into a latent semantic space, where their interactions can be effectively inferred and fused. This relational modeling is crucial for correcting noisy or uncertain predictions and enhancing localization robustness under online constraints.

To this end, we design two specialized attention mechanisms: TCA and TEA, which operate on the actor distribution maps to extract inter-distribution semantics and produce refined spatial priors.

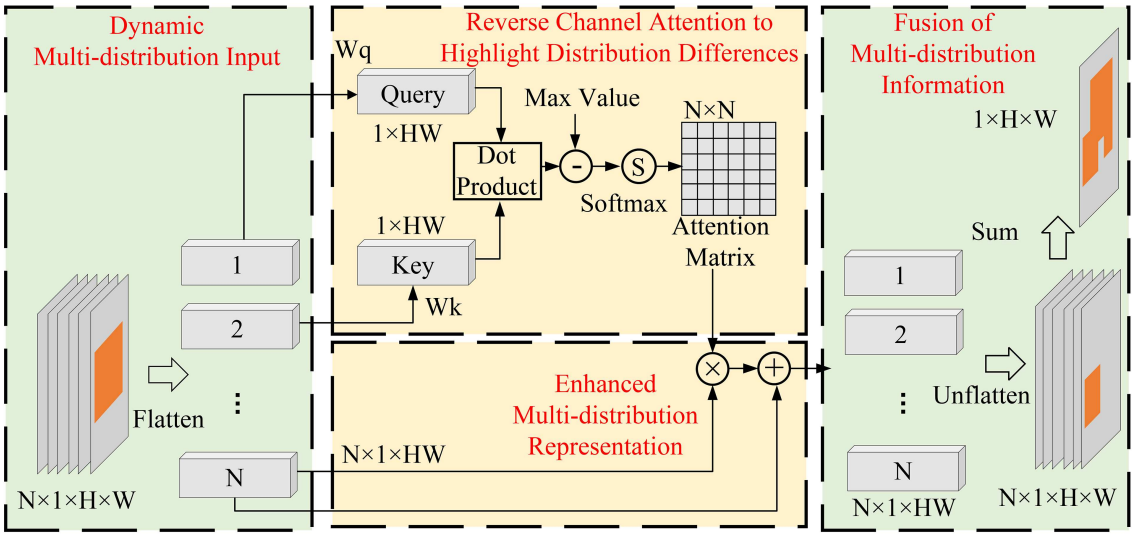

Figure 3 (Color online) Illustration of the track channel attention (TCA) mechanism. Each predicted actor distribution is independently flattened and projected into a high-dimensional semantic space through learnable matrices W_q and W_k, generating query and key representations. An attention matrix is computed to model semantic similarity between distributions. A reverse attention scheme is applied to emphasize inter-distribution differences, and enhanced vectors are fused and reshaped to form a final aggregated actor distribution map.

Formally, given N predicted bounding boxes at the current frame, we generate N single-target distribution maps, where each pixel value represents the spatial confidence of the corresponding actor. These maps are stacked along the channel dimension to construct a multi-channel actor distribution map $\mathbf{H} \in \mathbb{R}^{N\times H\times W}$.

3.2.1 *Tracking channel attention*

Conventional channel attention mechanisms typically operate under the assumption of fixed channel dimensions and are designed for dense, static feature maps. However, in our scenario, the number of predicted actor distributions varies dynamically with the number of actors, resulting in a multi-channel distribution map $\mathbf{H}$ with a non-fixed number of channels. To address this, we introduce TCA, a lightweight mechanism that infers semantic relations across dynamically varying actor distributions, enabling adaptive and robust information fusion.

To handle the dynamically changing number of channels in $\mathbf{H}$, we flatten each channel into a vector $\boldsymbol{h}_i$ and project it to a semantic space using learnable weights $\mathbf{W}_q$ and $\mathbf{W}_k$. Attention weights between channels are computed as

$$A_{i,j} = \frac{\exp(\mathbf{W}_q\boldsymbol{h}_i \cdot \mathbf{W}_k\boldsymbol{h}_j)}{\sum_{p=1}^{N}\exp(\mathbf{W}_q\boldsymbol{h}_i \cdot \mathbf{W}_k\boldsymbol{h}_p)}. \tag{4}$$

Each vector is then updated with a weighted sum of others:

$$\boldsymbol{h}_i' = \boldsymbol{h}_i + \sum_{j=1}^{N} A_{i,j}\boldsymbol{h}_j. \tag{5}$$

The updated vectors are reshaped and summed across channels to yield the final actor distribution map:

$$H'_{m,n} = \sum_{i=1}^{N} H_{i,m,n}. \tag{6}$$

This mechanism is designed to extract semantic relationships across dynamically varying channels and enhance information fusion among single-target distributions. The architecture of TCA is illustrated in Figure 3.

3.2.2 *Tracking efficient attention*

Conventional spatial attention mechanisms are typically designed for feature maps with fixed channel counts, dense activations, and compact target regions. However, the actor distribution maps in our setting present the opposite characteristics: they have a small and dynamically varying number of channels, sparse activations focused on actor locations, and target regions that often occupy a large proportion of the spatial map. These differences make

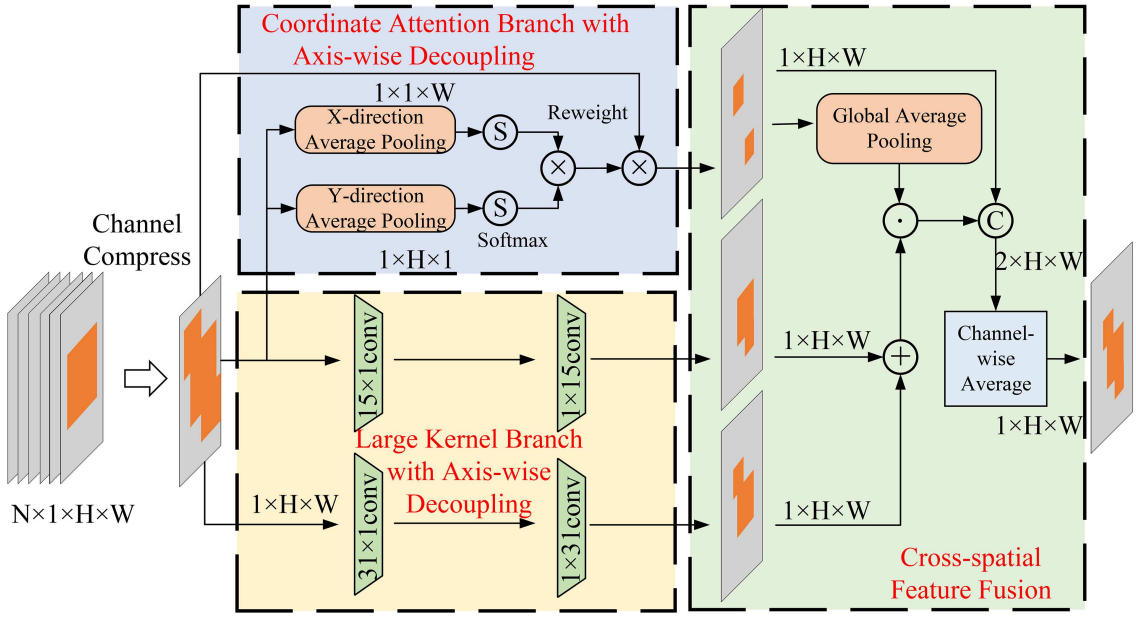

Figure 4 (Color online) Illustration of the track efficient attention (TEA) module. The module performs coordinate-aware attention and multi-scale enhancement using axis-wise decoupling and large kernel branches, followed by cross-spatial feature fusion.

standard attention designs less effective. To this end, we propose TEA, which integrates multi-scale large-kernel convolutions and coordinate-aware attention to robustly model semantic cues under such spatial and structural conditions.

As shown in Figure 4, we first compress $\mathbf{H}$ along the channel dimension. The resulting single-channel map is then processed by multiple enhancement branches.

Coordinate attention branch applies global pooling in horizontal and vertical directions to produce attention vectors $\boldsymbol{x}_{\text{ca}}$ and $\boldsymbol{y}_{\text{ca}}$, which are combined into a spatial attention map:

$$\mathbf{A}_{\text{ca}} = \sigma(\boldsymbol{x}_{\text{ca}}) \cdot \sigma(\boldsymbol{y}_{\text{ca}})^{\top}. \tag{7}$$

Large-kernel convolution branches use multiple convolutions with varying kernel sizes to extract multi-scale semantic features. Their outputs are aggregated and fused with the coordinate attention output, producing the final actor distribution map.

This strategy improves both the spatial precision and semantic richness of the predicted actor distribution, providing reliable priors for the subsequent modulation process.

3.3 History-aware feature modulation strategy

After actor distributions are inferred via track attention, we obtain a refined distribution map that provides spatial priors for actor locations at the current moment. To normalize confidence values, we apply a sigmoid function to constrain all values within the range $[0, 1]$. This distribution map is then used to modulate the action features, enhancing regions that align with predicted actor trajectories while suppressing irrelevant background areas.

To bridge the structural and semantic gap between the actor distribution and action features, we propose a history-aware feature modulation strategy, as illustrated in Figure 5. This strategy dynamically generates channel-wise attention weights by learning semantic correlations between the actor distribution and each feature channel. These weights guide the selective enhancement or suppression of both spatial regions and feature channels, effectively injecting reliable historical priors into the feature representation.

The process comprises three steps: spatial alignment, channel attention weight estimation, and feature modulation.

Spatial alignment. The predicted actor distribution map $\mathbf{H} \in \mathbb{R}^{H \times W}$ is downsampled via pooling to match the spatial resolution of the action feature map $\mathbf{D} \in \mathbb{R}^{C \times H \times W}$.

Channel attention weight estimation. We flatten each feature channel $\boldsymbol{d}_i$ and the distribution map $\boldsymbol{h}$, and project them into a shared semantic space using learnable matrices $\mathbf{W}_q$ and $\mathbf{W}_k$. The cross-attention weight a_i for

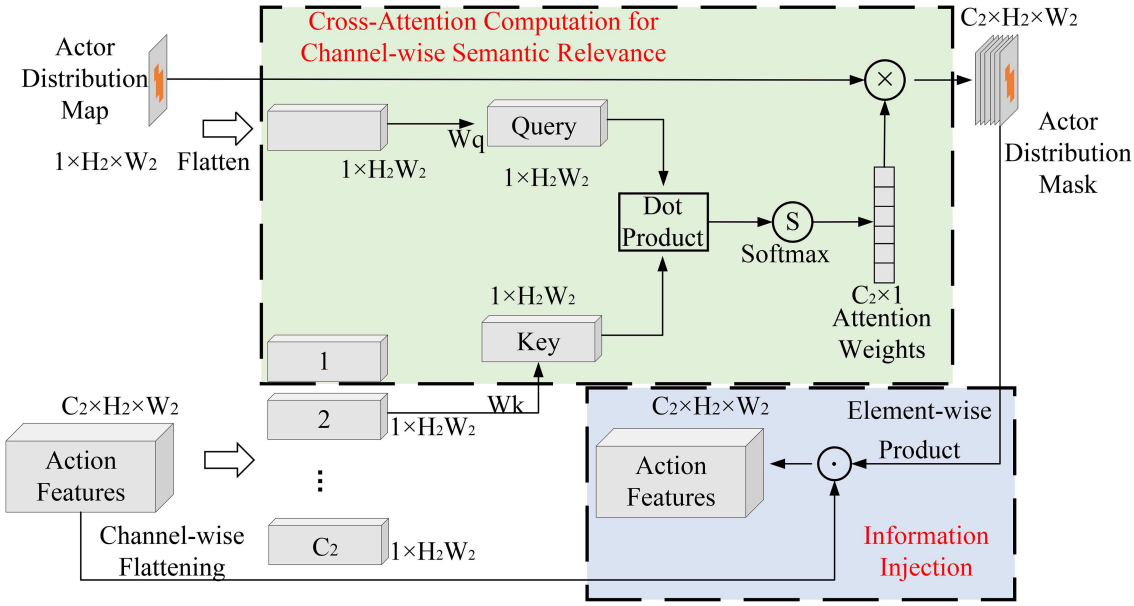

Figure 5 (Color online) Illustration of the history-aware feature modulation strategy. Actor distribution priors guide the modulation of spatio-temporal action features through cross-attention and channel-wise reweighting.

each channel is computed as

$$a_i = \frac{\exp\left(\mathbf{W}_q \boldsymbol{h} \cdot \mathbf{W}_k \boldsymbol{d}_i\right)}{\sum_{p=1}^{C} \exp\left(\mathbf{W}_q \boldsymbol{h} \cdot \mathbf{W}_k \boldsymbol{d}_p\right)}, \tag{8}$$

where C is the number of feature channels.

Feature modulation. The modulated feature $\mathbf{D}'$ is computed by injecting actor distribution priors and channel-wise attention weights:

$$D'_{i,m,n} = (1-\rho) D_{i,m,n} + \rho \cdot a_i \cdot D_{i,m,n} \cdot H_{m,n}, \tag{9}$$

where ρ is a learnable blending coefficient, and $H_{m,n}$ is the confidence at spatial location (m,n) in the actor distribution map.

Through this mechanism, historical trajectory and distribution information are effectively fused into current action features, enhancing localization precision and improving detection performance under online constraints.

3.4 Loss functions

The overall objective of the TAN integrates multiple loss functions to jointly optimize tracking prediction and action detection performance. The total loss is defined as

$$L_{\text{total}} = \lambda_{\text{conf}} L_{\text{conf}} + \lambda_{\text{reg}} L_{\text{reg}} + \lambda_{\text{cls}} L_{\text{cls}} + \lambda_{\text{track}} L_{\text{track}}. \tag{10}$$

Tracking loss. To supervise the predicted actor distribution $\boldsymbol{h}$, we define a cosine similarity loss against the ground-truth actor distribution $\boldsymbol{h}_{\text{GT}}$:

$$L_{\text{track}} = 1 - \frac{\boldsymbol{h} \cdot \boldsymbol{h}_{\text{GT}}}{\|\boldsymbol{h}\| \|\boldsymbol{h}_{\text{GT}}\|}. \tag{11}$$

The ground-truth distribution $\boldsymbol{h}_{\text{GT}}$ is generated by fusing all ground-truth bounding boxes at each frame into a single-channel heatmap.

Confidence loss. A binary cross-entropy loss is applied to all N_{total} anchors to distinguish positive and negative detections:

$$L_{\text{conf}} = -\frac{1}{N_{\text{total}}} \sum_{j=1}^{N_{\text{total}}} \left(t_j^{\text{conf}} \log p_j^{\text{conf}} + (1 - t_j^{\text{conf}}) \log(1 - p_j^{\text{conf}}) \right), \tag{12}$$

where t_j^{conf} is the binary ground-truth confidence label.

Regression loss. For each of the N_{pos} positive samples, a GIoU-based loss is applied to optimize bounding box localization:

$$L_{\text{reg}} = \frac{1}{N_{\text{pos}}} \sum_{j=1}^{N_{\text{pos}}} \left(1 - \text{GIoU}(b_j, t_j^{\text{reg}})\right), \tag{13}$$

where b_j and t_j^{reg} denote the predicted and target bounding boxes, respectively.

Classification loss. For action classification, we apply binary cross-entropy across N_c action classes:

$$L_{\text{cls}} = -\frac{1}{N_{\text{pos}}} \sum_{j=1}^{N_{\text{pos}}} \sum_{k=1}^{N_c} t_{j,k}^{\text{cls}} \log p_{j,k}^{\text{cls}}, \tag{14}$$

where $p_{j,k}^{\text{cls}}$ is the predicted probability for class k and $t_{j,k}^{\text{cls}}$ is the ground truth.

4 Experiments

4.1 Experimental setup

We evaluate the proposed TAN on two densely annotated benchmarks: JHMDB21 [33] and UCF24 [34]. In contrast to popular spatio-temporal action detection datasets such as AVA, which adopt sparse temporal annotations and are less suitable for online detection tasks, JHMDB21 and UCF24 provide dense frame-level annotations, making them ideal for evaluating OSTAD methods that require per-frame predictions without accessing future frames.

Video segment sampling. During training, each video segment consists of 16 frames sampled from past to present. The sampling stride is randomly selected from {1, 2} to increase temporal variation and robustness to different motion speeds. During inference, the stride is fixed to 1 to ensure temporal consistency. All frames are resized to 224×224 before being input to the network.

Feature map configuration. The feature extraction backbone generates multi-scale feature maps at three levels. The corresponding spatial resolutions are $H_1 = 112$, $H_2 = 28$, and $H_3 = 14$, while the channel dimensions are $C_1 = 64$, $C_2 = 256$, and $C_3 = 2048$. The channel configuration adopts a progressive expansion strategy, starting from low-dimensional early features to high-dimensional deeper layers, following established practices such as in SlowFast [35]. This design facilitates the extraction of both low-level patterns (e.g., edges, motion) and high-level semantics (e.g., actor interactions, action context), ensuring effective multi-scale representation under online constraints. The spatial resolutions are chosen to match common resolution reduction schemes while preserving sufficient localization precision for actor-level detection.

Loss composition. The total training loss combines four components: classification loss L_{cls}, confidence loss L_{conf}, regression loss L_{reg}, and tracking loss L_{track}, with corresponding weights $\lambda_{\text{cls}} = 1$, $\lambda_{\text{conf}} = 1$, $\lambda_{\text{reg}} = 5$, and $\lambda_{\text{track}} = 1$. The weight configuration follows standard practices in modern detection frameworks. Initially, we applied equal weights to all components for balanced training. Through empirical evaluation, we increased the importance of L_{reg}, as precise bounding box localization is particularly critical in online scenarios, where false detections cannot be corrected post hoc.

Evaluation metrics. Following OSTAD conventions, we report both frame-level mAP (f-mAP) and video-level mAP (v-mAP) at an IoU threshold of 0.5. f-mAP evaluates detection accuracy at individual frames, while v-mAP assesses the temporal coherence of action tubes. We also include GFLOPs to reflect the computational cost of different model variants for fair comparison in real-time applications.

Implementation details. All experiments are implemented using PyTorch 1.8.0 with CUDA 11.2, conducted on a workstation with an Intel Xeon E5-2680 v4 CPU, dual NVIDIA RTX 3090 GPUs, and 128 GB RAM. The batch size is set to 8 and training follows a distributed data-parallel scheme with mixed-precision acceleration.

4.2 Comparison with existing methods

We compare the proposed TAN with a broad range of representative methods, including both mainstream offline STAD approaches and all available online STAD (OSTAD) methods. Unless otherwise specified, 16-frame input segments are used for all methods to ensure fairness. For methods supporting online inference, we report their online performance; for offline methods, we include their best reported results. Online methods are particularly noteworthy as they align with practical requirements for frame-by-frame incremental detection without accessing future frames. The comparison results are summarized in Tables 1 [4, 5, 7–13, 36–42] and 2 [3–5, 7, 9–11, 13–15, 36, 40, 43–45].

Table 1 Comparison with existing methods on JHMDB21. The best results are in bold.

Setting	Year	Method	GFLOPs	f-mAP@0.5	v-mAP@0.5
Offline	2016	MR-TS RCNN [36]	–	58.5	73.1
	2017	T-CNN [37]	–	61.3	76.9
	2019	STEP [38]	–	75.0	–
	2019	LSTR [39]	–	–	85.5
	2020	MOC [40]	–	70.8	77.2
	2020	CFAD [41]	–	–	85.3
	2022	TubeR [5]	240.0	–	82.3
	2023	SE-STAD [42]	–	–	82.5
	2023	EVAD [7]	116.3	**90.2**	77.8
Online	2015	Gkioxari et al. [8]	–	36.2	53.3
	2015	Weinzaepfel et al. [9]	–	45.8	60.7
	2016	Saha et al. [10]	–	–	71.5
	2017	ROAD [11]	–	–	72.0
	2018	ACRN [12]	–	–	80.1
	2019	YOWO [13]	27.7	70.7	82.3
	2024	YWOM [4]	–	73.3	83.7
	2024	TAN (TCA)	53.9	79.7	**88.3**
	2024	TAN (TEA)	55.4	**80.3**	88.0

Table 2 Comparison with existing methods on UCF24. The best results are in bold.

Setting	Year	Method	GFLOPs	f-mAP@0.5	v-mAP@0.5
Offline	2016	MR-TS RCNN [36]	–	65.7	–
	2017	AMTnet [43]	–	–	51.2
	2020	AIA [44]	–	78.8	–
	2020	MOC [40]	–	78.0	53.8
	2021	ACAR-Net [45]	–	84.3	–
	2022	TubeR [5]	240.0	83.2	**58.4**
	2023	EVAD [7]	116.3	85.1	58.8
	2024	SMAST [3]	–	85.5	–
Online	2015	Weinzaepfel et al. [9]	–	35.8	–
	2016	Saha et al. [10]	–	–	35.9
	2017	ROAD [11]	–	–	46.3
	2019	YOWO [13]	27.7	80.4	48.8
	2020	Zhang et al. [14]	–	67.7	46.6
	2023	YOWOv2 [15]	53.6	85.2	52.0
	2024	YWOM [4]	–	84.6	49.6
	2024	TAN (TCA)	53.9	86.9	54.7
	2024	TAN (TEA)	55.4	**88.1**	**54.8**

As shown in Table 1, even under the more challenging online setting, TAN achieves 80.3 f-mAP and 88.3 v-mAP at the IoU threshold of 0.5, outperforming all online methods and most offline methods. Compared to the best previous online baseline YWOM, TAN improves f-mAP by 9.5% (from 73.3 to 80.3) and v-mAP by 5.5% (from 83.7 to 88.3). Although EVAD [7] achieves a higher f-mAP, it uses up to 32 future frames per input, violating causal constraints, and adopts a heavyweight video transformer backbone (116.3 GFLOPs), making it unsuitable for real-time edge deployment. In contrast, TAN runs with only 55.4 GFLOPs and satisfies the constraints of online detection.

Table 2 further validates TAN's effectiveness on the UCF24 dataset. TAN achieves 88.1 f-mAP and 54.8 v-mAP, again outperforming all online baselines. Compared to YOWOv2, TAN (TEA) achieves gains of 3.4% f-mAP and 2.8% v-mAP. Although TubeR [5] and EVAD achieve slightly higher v-mAPs, they rely on long input sequences (up to 64 frames), future information, and complex transformers, which violate the principles of incremental inference and resource efficiency in online scenarios.

These improvements stem from TAN's history-aware design, which leverages spatio-temporal continuity, appearance cues, and trajectory predictions to refine current representations. By enriching current features with reliable priors from past observations, TAN enhances both detection precision and temporal consistency, making it a strong

Table 3 Effectiveness and efficiency of actor distribution inference strategies. The best results are in bold.

Mechanism	Correlation	Kernel	GFLOPs	JHMDB21		UCF24	
				f-mAP@0.5	v-mAP@0.5	f-mAP@0.5	v-mAP@0.5
TCA	Positive	–	53.9	75.7	85.5	85.3	50.2
TCA	Negative	–	53.9	79.7	**88.3**	86.9	54.7
TEA	–	3	52.0	72.3	82.0	83.9	50.8
TEA	–	15	53.0	76.2	85.6	85.1	51.1
TEA	–	31	54.2	79.5	87.3	87.2	52.3
TEA	–	15+31	55.4	**80.3**	88.0	**88.1**	**54.8**
–	–	–	50.1	72.5	79.3	82.2	47.3

fit for practical online deployment.

Under consistent input configurations, TAN (TCA) and TAN (TEA) achieve 29 and 27 FPS, respectively on an RTX 3090 GPU. This slight difference reflects the additional cost of TEA's large-kernel convolutions. Importantly, both variants operate well above the 25 FPS threshold commonly considered for real-time video processing, demonstrating that our model meets the online spatio-temporal action detection requirements.

4.3 Ablation studies

4.3.1 *Actor distribution inference*

To validate the effectiveness of the actor distribution inference strategy via track attention, we compare the performance and computational cost of different attention designs in this stage. In addition to our proposed TCA and TEA, we also include a baseline where the inference stage is removed entirely, and the predicted actor distribution from the previous stage is directly used for modulation. As shown in Table 3, both TCA and TEA significantly improve detection performance over the baseline across both datasets. On JHMDB21, TCA (79.7 f-mAP, 88.3 v-mAP) and TEA (80.3 f-mAP, 88.0 v-mAP) each achieve large gains over the no-inference setting (72.5 f-mAP, 79.3 v-mAP). The improvement is even more pronounced on UCF24, where TEA achieves 88.1 f-mAP and 54.8 v-mAP, demonstrating the value of semantic fusion among actor distributions in complex multi-actor scenarios. In the TCA design, we compare two correlation schemes when computing attention weights. The negatively correlated variant consistently outperforms the positively correlated one. This is likely because dissimilar actor distributions offer richer complementary information, and emphasizing such diversity benefits fusion robustness. Therefore, we adopt negative correlation as the default design in TCA. In the TEA design, we investigate the impact of the convolutional kernel size. While small 3×3 kernels result in limited receptive fields and inferior performance, larger kernels such as 15×15 and 31×31 offer better coverage of the actor distribution map and improve both frame-level and video-level accuracy. The best performance is achieved when combining both kernel sizes, forming a multi-scale representation that captures diverse spatial patterns. In terms of computational cost, TEA incurs slightly higher complexity (55.4 GFLOPs) compared to TCA (53.9 GFLOPs), due to the addition of large-kernel spatial convolutions. However, the improved accuracy (+1.2 f-mAP on UCF24) justifies the cost in many real-time applications. For resource-constrained settings, TCA still offers a competitive and lightweight alternative, requiring fewer spatial operations while still capturing meaningful inter-channel interactions. Overall, TEA provides a better trade-off between accuracy and efficiency for general use cases, while TCA serves as a more efficient solution when channel adaptivity is sufficient.

4.3.2 *Impact of history length, appearance feature, and EMA coefficient in actor distribution prediction*

We investigate the influence of the length of the historical detection window, appearance features, and the EMA coefficient α used for updating trajectory appearance embeddings. As shown in Table 4, extending the historical length up to 32 frames improves both frame- and video-level mAP on both datasets, as it provides more consistent motion and spatial cues. However, performance slightly declines when increasing to 64 frames due to the inclusion of outdated or noisy detections that weaken temporal causality and introduce irrelevant information. The use of appearance features further improves association accuracy, especially on UCF24, where complex scenes often contain multiple actors per frame. In contrast, JHMDB21 contains only one actor per frame, making the contribution of appearance features less significant. We also study the impact of the EMA coefficient α that controls the update rate of appearance features in actor trajectories. We observe that setting α to 0.9 yields optimal results across both datasets. Larger values (e.g., 0.95) reduce responsiveness to new appearance changes, while smaller values (e.g.,

Table 4 Impact of history length, appearance feature, and EMA coefficient in actor distribution prediction. The best results are in bold.

History length	Appearance	EMA (α)	JHMDB21		UCF24	
			f-mAP@0.5	v-mAP@0.5	f-mAP@0.5	v-mAP@0.5
8	✓	0.9	75.3	80.1	82.6	49.8
16	✓	0.9	79.1	87.5	86.9	53.7
32	✓	0.9	**80.3**	**88.0**	**88.1**	**54.8**
32	✓	0.8	79.4	87.4	87.6	54.1
32	✓	0.95	80.2	87.9	88.0	54.7
32	✓	0.7	78.6	85.8	86.4	53.1
64	✓	0.9	78.7	85.9	84.3	52.9
32	–	0.9	80.1	87.8	85.1	52.0

Table 5 Comparison of feature modulation strategies and impact of mixing ratio. The best results are in bold.

Modulation method	Mixing ratio ρ	JHMDB21		UCF24	
		f-mAP@0.5	v-mAP@0.5	f-mAP@0.5	v-mAP@0.5
Ours	0.1	76.3	83.1	84.7	50.2
Ours	0.2	80.1	87.7	**88.1**	**54.8**
Ours	0.3	**80.3**	**88.0**	**88.1**	**54.8**
Ours	0.5	80.0	87.5	86.9	54.7
Ours	0.7	77.9	84.6	83.1	50.9
Direct	0.3	75.7	81.9	81.6	49.3
Post-detection	–	73.4	80.3	79.7	47.5

Table 6 Comparison of cross-attention weight computation methods. The best results are in bold.

Weight computation	JHMDB21		UCF24	
	f-mAP@0.5	v-mAP@0.5	f-mAP@0.5	v-mAP@0.5
Dot product	**80.3**	**88.0**	**88.1**	**54.8**
Scaled dot product	79.2	87.3	86.9	53.9
Cosine similarity	75.4	83.2	81.1	49.7
Euclidean distance	78.6	86.1	84.9	53.0

0.7) make the model overly sensitive to short-term variations. This confirms that maintaining a stable but adaptive appearance representation is critical for reliable trajectory modeling.

4.3.3 *Effectiveness of history-aware feature modulation*

Table 5 compares the proposed modulation strategy with two alternatives. Our method adaptively weights each feature channel based on semantic alignment with the actor distribution, enabling both spatial and channel-aware refinement of action features. In contrast, direct modulation simply applies the actor distribution as a spatial mask across all feature channels equally, lacking semantic adaptivity. Post-detection modulation, on the other hand, directly filters detection outputs based on the actor distribution after the detection head, which may discard useful features too early and lacks feature-level refinement. We also evaluate the influence of the mixing ratio ρ, finding that values between 0.2 and 0.3 yield the best balance between preserving current features and injecting prior knowledge.

4.3.4 *Cross-attention in history-aware feature modulation strategy*

We evaluate four cross-attention mechanisms used in the proposed modulation strategy: dot product, scaled dot product, cosine similarity, and Euclidean distance. Table 6 shows that the simple dot product achieves the best performance, likely due to its ability to preserve both magnitude and direction of semantic vectors in high-dimensional space.

4.3.5 *Generalization to future-aware detection*

Finally, we test the flexibility of TAN by gradually incorporating future frames into the input segment during evaluation. Table 7 shows that while future information improves accuracy, the gain plateaus beyond 8 additional

Table 7 Complementarity of historical and future-aware inputs. The best results are in bold.

History frame	Future frame	JHMDB21		UCF24	
		f-mAP@0.5	v-mAP@0.5	f-mAP@0.5	v-mAP@0.5
15	0	80.3	88.0	88.1	54.8
15	4	83.3	88.6	88.6	55.0
15	8	85.5	**89.2**	89.7	56.2
15	16	**85.6**	**89.2**	**90.2**	**56.3**

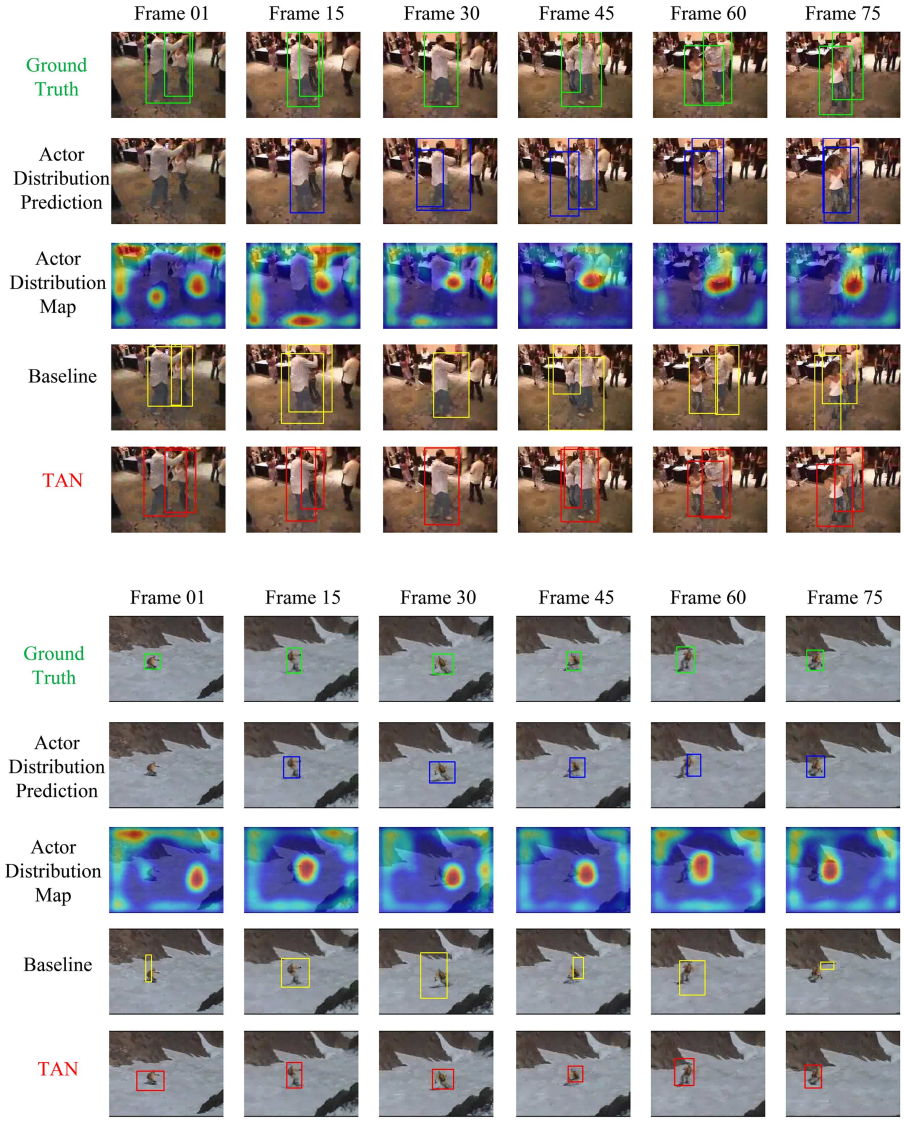

Figure 6 (Color online) Visualization of outputs at different stages of the proposed TAN. From top to bottom: ground truth, predicted distributions from actor tracking, attention heatmaps from track attention, detection results of the baseline, and detection results of TAN.

frames. This confirms the complementary role of historical and future context, and highlights TAN's ability to generalize to offline detection scenarios as well.

4.4 Visualization of detection results

To better understand the effectiveness of the proposed TAN, we visualize the outputs of different stages on two representative challenging actions: (1) salsa spin with frequent occlusion due to multi-person overlap, and (2) skiing with fast motion and small target scale. The visualizations are shown in Figure 6.

Each row shows a temporal sequence of frames, comparing ground truth (green boxes), predicted actor distri-

butions from the history-aware tracking module (blue), attention heatmaps from the track attention module (in red-yellow colormap), detection results of the baseline (yellow), and the final detection results of TAN (red). The baseline shares the same backbone and detection heads with TAN but excludes the tracking-guided refinement stage.

At the beginning of each video segment, both the baseline and TAN perform poorly. This is expected, as historical information has not yet accumulated, and tracking predictions are still inaccurate. The attention heatmaps also reflect this, with activations spread across non-target regions. For instance, in the first few frames of the salsa spin sequence, attention is diffused across the scene, failing to accurately localize the primary actor.

As the sequence progresses, TAN gradually benefits from the accumulation of reliable historical detections. The actor tracking prediction becomes more accurate, and the attention heatmaps begin to focus more precisely on the true target regions. Consequently, TAN achieves more stable and precise localization compared to the baseline. In particular, TAN shows superior robustness in frames with severe occlusion or significant displacement, where the baseline often fails.

These results validate the core design of TAN: historical information, when properly modeled through actor distribution prediction and track attention-based inference, provides strong priors for refining action features. This enables TAN to maintain high detection accuracy even in complex real-world scenarios with limited instantaneous visual information.

5 Conclusion

This paper addresses the challenge of insufficient effective information in OSTAD by proposing the TAN, a history-aware track-and-detect framework. TAN enhances current-frame detection through a tracking-guided refinement process, incorporating three core strategies: history-aware actor distribution prediction, actor distribution inference via track attention, and history-aware feature modulation.

Experiments on JHMDB21 and UCF24 demonstrate that TAN outperforms existing online methods and even surpasses several offline approaches, achieving up to 88.1 f-mAP and 88.3 v-mAP. Compared to the best online baselines, TAN improves average detection accuracy by 5.9%.

While effective, TAN's performance may still be affected by early detection errors and fixed hyperparameter settings. Future work includes developing adaptive mechanisms and extending TAN to model actor interactions and longer-term temporal dependencies for improved robustness in real-time applications.

References

1 Hu X, Dai J, Li M, et al. Online human action detection and anticipation in videos: a survey. Neurocomputing, 2022, 491: 395–413
2 Wang P, Zeng F, Qian Y. A survey on deep learning-based spatio-temporal action detection. 2023. ArXiv:2308.01618
3 Korban M, Youngs P, Acton S T. A semantic and motion-aware spatiotemporal transformer network for action detection. IEEE Trans Pattern Anal Mach Intell, 2024, 46: 6055–6069
4 Qin Y, Chen L, Ben X, et al. You watch once more: a more effective CNN architecture for video spatio-temporal action localization. Multimedia Syst, 2024, 30: 53
5 Zhao J, Zhang Y, Li X, et al. TubeR: tubelet transformer for video action detection. In: Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022
6 Zheng Y D, Chen G, Yuan M, et al. MRSN: multi-relation support network for video action detection. 2023. ArXiv:2304.11975
7 Chen L, Tong Z, Song Y, et al. Efficient video action detection with token dropout and context refinement. 2023. ArXiv:2304.08451
8 Gkioxari G, Malik J. Finding action tubes. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, 2014
9 Weinzaepfel P, Harchaoui Z, Schmid C. Learning to track for spatio-temporal action localization. In: Proceedings of IEEE International Conference on Computer Vision, 2015
10 Saha S, Singh G, Sapienza M, et al. Deep learning for detecting multiple space-time action tubes in videos. In: Proceedings of British Machine Vision Conference, 2016
11 Singh G, Saha S, Sapienza M, et al. Online real-time multiple spatiotemporal action localisation and prediction. In: Proceedings of IEEE International Conference on Computer Vision, 2017
12 Sun C, Shrivastava A, Vondrick C, et al. Actor-centric relation network. In: Proceedings of European Conference on Computer Vision, 2018
13 Kopuklu O, Wei X, Rigoll G. You only watch once: a unified CNN architecture for real-time spatiotemporal action localization. 2019. ArXiv:1911.06644
14 Zhang D, He L, Tu Z, et al. Learning motion representation for real-time spatio-temporal action localization. Pattern Recogn, 2020, 103: 107312
15 Yang J, Dai K. YOWOv2: a stronger yet efficient multi-level detection framework for real-time spatio-temporal action detection. 2023. ArXiv:2302.06848
16 Xu Y, Zhou X, Chen S, et al. Deep learning for multiple object tracking: a survey. IET Comput Vision, 2019, 13: 355–368
17 Ciaparrone G, Sánchez F L, Tabik S, et al. Deep learning in video multi-object tracking: a survey. Neurocomputing, 2020, 381: 61–88
18 Gao J, Zhang T, Xu C. Graph convolutional tracking. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019
19 Wan J X, Zhang H, Zhang J, et al. DSRRTracker: dynamic search region refinement for attention-based siamese multi-object tracking. 2022. ArXiv:2203.10729
20 Zhu Z, Hou J, Wu D O. Cross-modal orthogonal high-rank augmentation for RGB-event transformer-trackers. In: Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023
21 Chen X, Iranmanesh S M, Lien K C. Patchtrack: multiple object tracking using frame patches. 2022. ArXiv:2201.00080
22 Hassanin M, Anwar S, Radwan I, et al. Visual attention methods in deep learning: an in-depth survey. 2022. ArXiv:2204.07756

23 Hu J, Shen L, Sun G. Squeeze-and-excitation networks. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, 2018
24 Wang Q, Wu B, Zhu P, et al. ECA-Net: efficient channel attention for deep convolutional neural networks. In: Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020
25 Woo S, Park J, Lee J Y, et al. CBAM: convolutional block attention module. In: Proceedings of European Conference on Computer Vision, 2018
26 Hu X, Zhang Z, Jiang Z, et al. SPAN: spatial pyramid attention network for image manipulation localization. In: Proceedings of European Conference on Computer Vision, 2020
27 Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need. In: Proceedings of Advances in Neural Information Processing Systems, 2017
28 Hou R, Chang H, Ma B, et al. Cross attention network for few-shot classification. In: Proceedings of Advances in Neural Information Processing Systems, 2019
29 Li X, Chen Y, Liu J, et al. NAS-YOLOX: a SAR ship detection using neural architecture search and multi-scale attention. Connect Sci, 2023, 35: 1–32
30 Li J, Tang H, Li X, et al. LEF-YOLO: a lightweight method for intelligent detection of four extreme wildfires based on the YOLO framework. Int J Wildland Fire, 2024, 33: WF23044
31 Jiang Y, Yin G, Jing W, et al. Box-spoof attack against single object tracking. Appl Intell, 2024, 54: 1585–1601
32 Yu Y, Wan M, Qian J, et al. Feature selection for multi-label learning based on variable-degree multi-granulation decision-theoretic rough sets. Int J Approx Reason, 2024, 169: 109181
33 Jhuang H, Gall J, Zuffi S, et al. Towards understanding action recognition. In: Proceedings of IEEE International Conference on Computer Vision, 2013
34 Soomro K, Zamir A M, Shah M. UCF101: a dataset of 101 human actions classes from videos in the wild. 2012. ArXiv:1212.0402
35 Feichtenhofer C, Fan H, Malik J, et al. Slowfast networks for video recognition. In: Proceedings of IEEE/CVF International Conference on Computer Vision, 2019
36 Peng X, Schmid C. Multi-region two-stream R-CNN for action detection. In: Proceedings of European Conference on Computer Vision, 2016
37 Hou R, Chen C, Shah M. Tube convolutional neural network (T-CNN) for action detection in videos. In: Proceedings of IEEE International Conference on Computer Vision, 2017
38 Yang X, Yang X, Liu MY, et al. STEP: spatio-temporal progressive learning for video action detection. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, 2019
39 Li D, Yao T, Qiu Z, et al. Long short-term relation networks for video action detection. In: Proceedings of the ACM International Conference on Multimedia, 2019
40 Li Y, Wang Z, Wang L, et al. Actions as moving points. In: Proceedings of European Conference on Computer Vision, 2020
41 Li Y, Lin W, See J,et al. CFAD: coarse-to-fine action detector for spatiotemporal action localization. In: Proceedings of European Conference on Computer Vision, 2020
42 Sui L, Zhang C L, Gu L, et al. A simple and efficient pipeline to build an end-to-end spatial-temporal action detector. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, 2023
43 Saha S, Singh G, Cuzzolin F. AMTnet: action-micro-tube regression by end-to-end trainable deep architecture. In: Proceedings of IEEE International Conference on Computer Vision, 2017
44 Tang J, Xia J, Mu X, et al. Asynchronous interaction aggregation for action detection. In: Proceedings of European Conference on Computer Vision, 2020
45 Pan J, Chen S, Shou M Z, et al. Actor-context-actor relation network for spatio-temporal action localization. In: Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021