TrackNetV4: Enhancing Fast Sports Object Tracking with Motion Attention Maps

Arjun Raj
School of Computing
Australian National University
Canberra, Australia
u7526852@anu.edu.au

Lei Wang*
School of Computing
Australian National University
Canberra, Australia
lei.w@anu.edu.au

Tom Gedeon
School of Elec Eng, Comp & Math Sci
Curtin University
Perth, Australia
tom.gedeon@curtin.edu.au

***Abstract*—Accurately detecting and tracking high-speed, small objects, such as balls in sports videos, is challenging due to factors like motion blur and occlusion. Although recent deep learning frameworks like TrackNetV1, V2, and V3 have advanced tennis ball and shuttlecock tracking, they often struggle in scenarios with partial occlusion or low visibility. This is primarily because these models rely heavily on visual features without explicitly incorporating motion information, which is crucial for precise tracking and trajectory prediction. In this paper, we introduce an enhancement to the TrackNet family by fusing high-level visual features with learnable motion attention maps through a motion-aware fusion mechanism, effectively emphasizing the moving ball's location and improving tracking performance. Our approach leverages frame differencing maps, modulated by a motion prompt layer, to highlight key motion regions over time. Experimental results on the tennis ball and shuttlecock datasets show that our method enhances the tracking performance of both TrackNetV2 and V3. We refer to our lightweight, plug-and-play solution, built on top of the existing TrackNet, as TrackNetV4.**

***Index Terms*—tracking, motion attention, fusion**

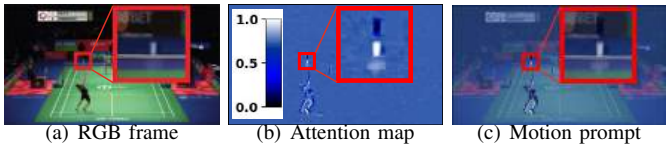
I. Introduction

Ball trajectory data is a crucial element in sports analysis and athlete training. However, accurately detecting and tracking high-speed, small balls in sports competition videos presents significant challenges. The primary difficulties arise from the fact that balls in broadcast videos often appear blurry, tiny, or obscured by afterimages. Additionally, they may become invisible due to occlusion, extreme visual indistinctness, or simply flying out of the camera's field of view.

With recent advances in deep learning, TrackNetV1 is introduced in [6] to track tennis balls and shuttlecocks in broadcast match videos. This heatmap-based tracking framework, built on a VGG-16 feature extraction network [9] and an upsampling network [8], takes multiple consecutive frames as input, leveraging the ball's trajectory[1] for improved detection. While TrackNetV1 achieves superior tracking performance compared to conventional methods, its processing speed is insufficient for real-time sports analysis, and its network design consumes substantial GPU memory.

TrackNetV2, presented in [10], offers improvements over TrackNetV1 by: (i) increasing processing speed through reduced input size (*e.g.*, from 640×360 to 512×288) and a redesigned multiple-in, multiple-out architecture (*e.g.*, from 3 in 1 out to 3 in 3 out), (ii) enhancing tracking accuracy by introducing and training on a comprehensive badminton match video dataset, and (iii) lowering GPU memory usage by replacing a pixel-wise one-hot encoding 3D array with a real-valued 2D array. Additionally, TrackNetV2 introduces a weighted cross-entropy loss function to focus on ball movements more effectively, and uses skip connections to preserve tiny object information, preventing the degradation of small object features in the network. Experiments show that TrackNetV2 significantly improves both trajectory prediction and real-time processing speed for badminton tracking. Similar to TrackNetV1, TrackNetV2 still does not explicitly consider motion. Its tracking and prediction performance are enhanced through a few skip connections that fuse low- and high-level feature maps extracted solely from video frames.

Recently, TrackNetV3 [3] further improved ball tracking accuracy and trajectory completeness, even when the ball is temporarily obstructed. Compared to TrackNetV2, TrackNetV3 incorporates estimated background information as auxiliary data to better locate the ball and introduces a trajectory rectification module that interpolates missing ball coordinates using image inpainting techniques. TrackNetV3 surpasses baselines like TrackNetV2 and YOLOv7 [11] by implicitly using motion dynamics, for example, through the subtraction between the original video frame and the estimated



(a) RGB frame (b) Attention map (c) Motion prompt

Fig. 1. A visual comparison is presented between (a) the original video frame, (b) the learned motion attention map, and (c) the motion-prompted frame. Tracking shuttlecocks is challenging due to their small size and tendency to blend into the background. To address this, we use a motion prompt layer [1] to generate motion attention maps that highlight the shuttlecock's location. We also create a motion-prompted frame by performing element-wise multiplication between the motion attention map and the original video frame, showing how motion features enhance visual representation. For better visualization, the shuttlecocks in these frames are zoomed in on the right.

* Corresponding author. Our project website is here.

[1]The ball's trajectory is determined by a sequence of video frames, yet this process does not explicitly incorporate any form of motion cues.

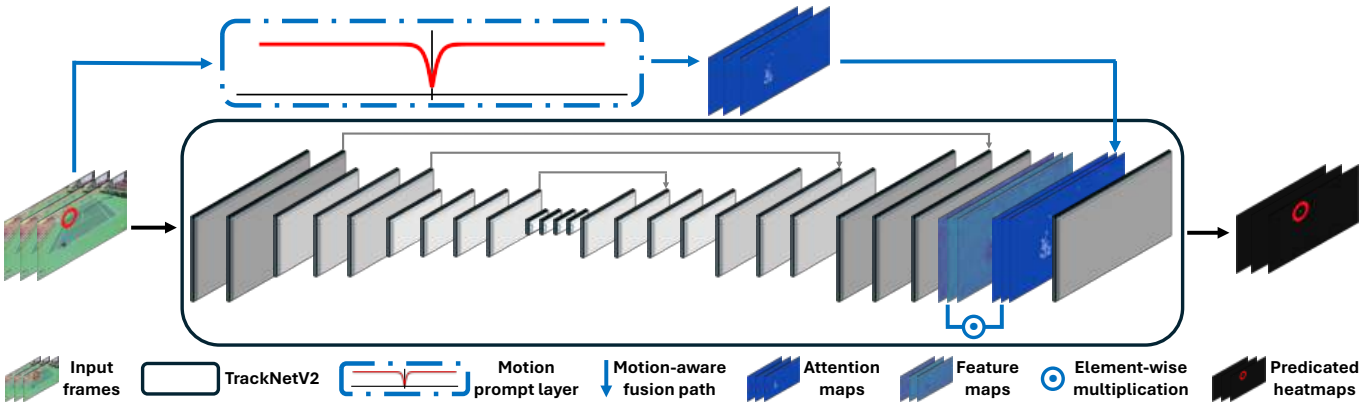


Fig. 2. We propose using learnable motion attention maps to enhance the tracking of high-speed, small objects in video frames. While demonstrated with TrackNetV2, our approach can be seamlessly integrated into any heatmap-based detection and tracking framework. Our method uses a motion prompt layer [1] on frame differencing maps (using absolute values to capture both positive and negative pixel intensity changes, thereby reducing missed detections) to generate motion attention maps that highlight key motion regions, such as balls. These maps are then fused with high-level visual features before the heatmap output layer through element-wise multiplication, followed by concatenation. The tracking framework that features our motion-aware fusion is named TrackNetV4.

background image. However, it still struggles when the ball is barely visible, relying heavily on temporal dynamics from consecutive frames, which often contain noise and irrelevant motion information.

Ball tracking and prediction in sports analysis are highly dependent on effective temporal dynamics extraction, yet even the state-of-the-art TrackNetV3 does not explicitly leverage motion information. Inspired by the recent success of prompts in computer vision tasks and the use of motion as learnable prompts [1] to selectively focus on relevant motions while suppressing unwanted noise, we propose a simple yet innovative module that uses learnable motion attention maps to enhance the tracking. Specifically, our approach: (i) selectively highlights motion regions of interest in video frames, and (ii) introduces motion-aware fusion mechanism to preserve the motion regions of high-speed, small objects.

Our contributions are summarized as follows:

i. We are the first to incorporate learnable motion attention maps into the tracking framework, enabling it to focus on movements crucial for accurately tracking small objects.
ii. We introduce a motion-aware fusion mechanism that combines motion attention maps with high-level visual features through element-wise multiplication, significantly improving tracking performance.
iii. We show that integrating motion concepts with a simple, plug-and-play fusion module into TrackNetV2 and TrackNetV3 enhances the tracking of fast-moving, small objects, resulting in our improved model, TrackNetV4.

This paper is organized as follows: Sec. II introduces our motion-aware fusion framework, Sec. III shows our experiments and discussions, and Sec. IV concludes the paper.

II. Approach

A. Preliminary

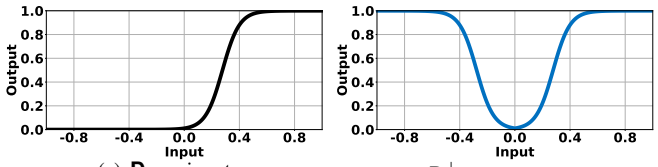
Notation. Let $\mathcal{I}_T$ denote the index set $1, 2, \cdots, T$. We use regular fonts for scalars (*e.g.*, x), lowercase boldface letters (*e.g.*, $\boldsymbol{x}$) for vectors, uppercase boldface letters (*e.g.*, $\boldsymbol{X}$) for matrices, and calligraphic letters (*e.g.*, $\mathbf{X}$) for tensors. Let $\mathbf{X} \in \mathbb{R}^{d_1 \times d_2 \times d_3}$ denote a third-order tensor. Using Matlab notation, we refer to its t-th slice as $\mathbf{X}_{:,:,t}$, which is a $d_1 \times d_2$ matrix.

Motivation. Prompts can extend beyond text or signals; they can also be learnable [12], [16] and take various forms [2], [4], [5], [7], [13]–[15]. Recent work [1] introduces a motion prompt layer with only two learnable parameters that modulate frame differencing maps to produce motion attention maps. These maps spatially highlight regions where motion is relevant (*e.g.*, balls) and suppress irrelevant motion (*e.g.*, video noise and background movement), while also capturing the temporal evolution of these attention maps over time. We explore the application of motion attention maps in tracking and predicting high-speed, small objects in professional ball games (*e.g.*, badminton, tennis, football, golf, *etc.*).

B. Motion Attention Maps and Motion-Aware Fusion

Fig. 2 provides an overview of our motion-aware fusion framework. For simplicity, we use TrackNetV2 for visualization, but our motion-aware fusion can be seamlessly integrated into any existing heatmap-based detection and tracking framework. Our approach offers a straightforward yet effective solution: (i) using a motion prompt layer to highlight relevant motions as attentions, and (ii) fusing the motion attention maps with high-level visual feature maps to preserve both the visual and motion information of small objects, thereby enhancing detection and tracking. Following best practices in [3], [10], we also use a multiple-input, multiple-output design.

Learnable attention maps for highlighting motion. Given a video sequence $\mathbf{X} = [\mathbf{F}_1, \mathbf{F}_2, \cdots, \mathbf{F}_T] \in \mathbb{R}^{H \times W \times 3 \times T}$, where $\mathbf{F}_t$ ($t \in \mathcal{I}_T$) represents the t-th frame, and H and W denote the frame height and width, respectively, we select T' as the number of input frames to form short-term temporal blocks. The t-th temporal block is represented as $\mathbf{X}_t = [\mathbf{F}_t, \mathbf{F}_{t+1}, \cdots, \mathbf{F}_{T'+t-1}] \in \mathbb{R}^{H \times W \times 3 \times T'}$ for $t \in \mathcal{I}_{T-T'+1}$.



(a) $\mathbf{D}_t$ as input (b) $\mathbf{D}_t^+$ as input (ours)

Fig. 3. Comparison between (a) original frame differencing maps $\mathbf{D}_t$ and (b) absolute frame differencing maps $\mathbf{D}_t^+$, both using the same normalization function from [1] with a slope of 16.24 and a shift of 0.28 for visualization. Our approach captures both positive and negative intensity changes, ensuring key motions for tracking and prediction are not missed, unlike [1], which maps negative values to 0.

Each temporal block is then converted into a grayscale video sequence, $\mathbf{X}'_t = [\boldsymbol{F}'_t, \boldsymbol{F}'_{t+1}, \cdots, \boldsymbol{F}'_{T'+t-1}] \in \mathbb{R}^{H\times W\times T'}$. After normalizing pixel values between 0 and 1, frame differencing maps are computed between consecutive frames, resulting in $\mathbf{D}_t = [\boldsymbol{D}_t, \boldsymbol{D}_{t+1}, \cdots, \boldsymbol{D}_{T'+t-2}] \in \mathbb{R}^{H\times W\times (T'-1)}$, where $\boldsymbol{D}_t = \boldsymbol{F}'_{t+1} - \boldsymbol{F}'_t$ ($t \in \mathcal{I}_{T'-1}$). Positive values in $\boldsymbol{D}_t$ indicate an increase in pixel intensity from frames t to $t+1$, while negative values indicate a decrease. Since $\mathbf{D}_t$ includes both positive and negative pixel intensity changes, such as minor object movements, we take the absolute values to capture all relevant motions for tracking and prediction, resulting in $\mathbf{D}_t^+$ ranging between 0 and 1. Fig. 3 shows the difference between using original frame differencing maps and absolute frame differencing maps. We apply a Power Normalization (PN) function a with learnable parameters $\boldsymbol{\theta}$ as in [1] to these differencing maps, producing a sequence of motion attention maps for the t-th temporal block:

$$\mathbf{A}_t = a_{\boldsymbol{\theta}}(\mathbf{D}_t^+), \tag{1}$$

where $t \in \mathcal{I}_{T'-1}$ and $\mathbf{A}_t \in \mathbb{R}^{H\times W\times (T'-1)}$.

Fusing motion attention maps with visual features. Inspired by the performance gains achieved through skip connections that address the gradual loss of tiny object features along the processing pipeline (*e.g.*, in TrackNetV2), we introduce a specialized motion-aware fusion mechanism. This mechanism integrates high-level visual features with our motion attention maps, preserving the ball's location and trajectory.

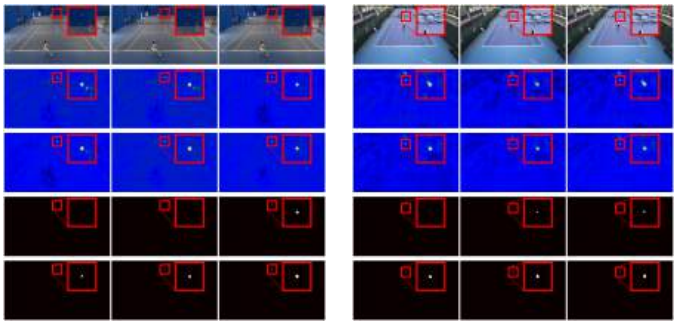
Specifically, we first extract high-level visual feature maps using the tracking network up to the last convolutional block (just before the Sigmoid layer that outputs the heatmaps):

$$\mathbf{V}_t = \text{TrackNet}_{\text{visual}}(\mathbf{X}_t), \tag{2}$$

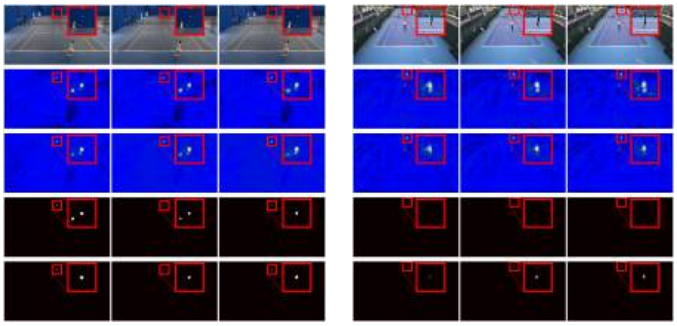
where $\text{TrackNet}_{\text{visual}}(\cdot)$ refers to the TrackNet for extracting the visual features $\mathbf{V}_t = [\boldsymbol{V}_t, \boldsymbol{V}_{t+1}, \cdots, \boldsymbol{V}_{T'+t-1}] \in \mathbb{R}^{H\times W\times T'}$. We then aggregate these visual feature representations with the motion attention maps generated via Eq. (1) from the motion prompt layer:

$$\mathbf{H}_t = \sigma(\mathbf{A}_t \circledcirc \mathbf{V}_t), \tag{3}$$

where $\sigma(\cdot)$ is the Sigmoid function, and $\mathbf{H}_t \in \mathbb{R}^{H\times W\times T'}$ denotes the motion-attention-enhanced heatmaps. The symbol



(a) Improved feature maps. (b) Clearer ball predictions.



(c) Refined ball localization. (d) Reduced missed detections.

Fig. 4. Comparison of feature maps and heatmaps with and without motion-aware fusion. Four visualization groups are shown, with the first row in each displaying the original frames. Motion-aware fusion improves visual representations (*e.g.*, 2nd vs. 3rd row in (a)), resulting in clearer, more accurate ball predictions ((a) and (c)). Combined with high-level features, motion attention further refines ball localization (*e.g.*, 4th vs. 5th row in (c)), reducing missed detections compared to the baseline ((b) and (d)). This demonstrates how motion awareness enhances tracking of fast-moving, small objects.

$\circledcirc$ represents our fusion operation, which involves element-wise multiplication followed by concatenation:

$$\mathbf{A}_t \circledcirc \mathbf{V}_t = [\boldsymbol{V}_t, \boldsymbol{A}_t \odot \boldsymbol{V}_{t+1}, \cdots, \boldsymbol{A}_{T'+t-2} \odot \boldsymbol{V}_{T'+t-1}], \tag{4}$$

where $\boldsymbol{A}_\tau \odot \boldsymbol{V}_{\tau+1}$ ($\tau \in \mathcal{I}_{T'+t-2}$) denotes the motion-enhanced visual representations, and $\mathbf{A}_t \circledcirc \mathbf{V}_t \in \mathbb{R}^{H\times W\times T'}$. As shown in Eq. (3), these representations are then passed through a Sigmoid function to produce the final heatmaps, which highlight the ball's location and trajectory over time.

We name the tracking framework incorporating our motion-aware fusion mechanism TrackNetV4. Below, we present our experimental results, comparisons and discussions.

III. Experiment

Dataset, protocol, and setup. The tennis ball tracking dataset introduced in [6] lacks a standardized training and test split, as it is initially divided randomly. This random division has led to inconsistencies across different works, making it difficult to fairly compare the results with other models. To address this issue, we develop two evaluation protocols while maintaining the 70/30 frame split:

i. *Game-level*: The training set includes 'game5', 'game10', 'game6', 'game2', 'game7', 'game3', and 'game8', , while the test set comprises 'game1', 'game9', and 'game4'. This

TABLE I

PERFORMANCE AND SPEED COMPARISONS OF BASELINES VERSUS TRACKNETV4 (BASELINES WITH OUR MOTION-AWARE FUSION, DENOTED AS +MOTION) ON THE SHUTTLECOCK AND TENNIS BALL TRACKING DATASETS. FOR TENNIS BALL TRACKING, WE APPLY OUR PREDEFINED (I) GAME-LEVEL AND (II) CLIP-LEVEL EVALUATION PROTOCOLS. THE "TOTAL" COLUMN INDICATES THE TOTAL NUMBER OF FRAMES USED FOR EVALUATION. FOR TRACKNETV2, A 3-IN 3-OUT SETUP IS USED. THE ROWS HIGHLIGHTED IN BLUE DENOTE OUR TRACKNETV4.

	Method	Total	Confusion matrix					Performance				Speed
			TP	TN	FP1	FP2	FN	Acc.	Prec.	Rec.	F1	
Tennis ball (i)	TrackNetV2	17193	15863	396	142	17	775	94.6	**99.0**	95.3	97.1	156.9
	TrackNetV2 (+Motion)	17193	15973	389	167	24	640	**95.2**	98.8	**96.1**	**97.5**	155.7
Tennis ball (ii)	TrackNetV2	17769	16195	393	163	25	993	93.4	**98.9**	94.2	96.4	160.9
	TrackNetV2 (+Motion)	17769	16374	399	199	19	778	**94.4**	98.7	**95.5**	**97.0**	158.6
Shuttlecock	YOLOv7	-	-	-	-	-	-	57.8	78.5	60.0	68.0	-
	TrackNetV2 (3 in 1 out)†	13064	9447	1514	751	218	1134	83.9	90.7	89.2	89.9	*12.9*
	TrackNetV2 (3 in 3 out)†	39192	29129	4264	468	358	4973	85.2	97.2	85.4	90.9	*31.8*
	TrackNetV2	37794	26324	6013	438	493	4526	85.6	**96.6**	85.3	90.6	163.3
	TrackNetV2 (+Motion)‡	37794	26592	5995	523	511	4173	**86.2**	96.3	**86.4**	**91.1**	139.1
	TrackNetV2 (+Motion)	37794	26878	5834	765	672	3645	**86.6**	94.9	**88.1**	**91.4**	161.1
	TrackNetV3	10836	8980	1395	22	8	431	95.7	**99.7**	95.4	97.5	15.1*
	TrackNetV3 (+Motion)	10836	9050	1400	30	10	346	**96.4**	99.5	**96.3**	**97.9**	15.1*

† indicates results from the original TrackNetV2 paper [10]. ‡ indicates the results obtained by fine-tuning the pretrained baseline TrackNetV2 for 3 epochs. * indicates the processing speed of the entire script, including data loading, file writing, *etc*. This may not be directly comparable to the other speeds.

division results in 70.81% of the total frames being used for training and 29.19% for testing.

ii. *Clip-level*: The dataset, originally organized into several clips per game, is split based on cumulative frames to ensure that 70% of the total frames are allocated for training. Each clip is assigned entirely to either the training or test set, maintaining disjoint training and test sets.

Additionally, we train and test our framework on the shuttlecock dataset introduced in [10], following the standard evaluation protocol. For all evaluations, we use the same loss functions and parameter setups [3], [10] as the original authors, including training epochs and learning rates.

Evaluation metrics. We evaluate our framework's performance by measuring the distance between the predicted and actual positions of the ball. Following the criteria set in [3], [10], a detection is considered accurate if this distance is within 4 pixels; otherwise, it is deemed inaccurate. Additionally, any inconsistency in detecting the ball's presence or absence is also classified as inaccurate. Beyond accuracy, we report precision, recall, and F1 score to provide a comprehensive evaluation. To assess the efficiency of our framework, we also measure the frames per second (FPS).

Qualitative results. Fig. 4 presents a visual comparison of (i) standard visual feature maps from the baseline (TrackNetV2) versus motion-enhanced feature maps obtained by applying our method (using element-wise multiplication as described in Eq. (4)), and (ii) heatmaps generated by the baseline (TrackNetV2) compared to those produced by our TrackNetV4 (TrackNetV2 + motion-aware fusion).

We observe that the motion-aware fusion significantly enhances the tracking and prediction of ball locations, as demonstrated by the clearer visualizations of both the feature maps (before applying the Sigmoid function) and the generated heatmaps. Notably, our feature maps exhibit greater clarity, and the resulting heatmaps demonstrate increased robustness. This highlights the effectiveness of our method, which remains lightweight while successfully tracking high-speed, small objects in sports scenarios.

Quantitative results. Table I summarizes our results on both the tennis ball and shuttlecock datasets. We denote the use of our motion-aware fusion as '(+Motion)' in the table. As shown, TrackNetV4 (applying motion attention maps to TrackNetV2) consistently improves performance, particularly in accuracy and F1-score metrics, while also reducing the number of false negatives in both tennis ball evaluation protocols.

We also observe that, in general, protocol (i) game-level evaluations outperform protocol (ii) clip-level evaluations across all four metrics by more than 0.5%. This suggests that models trained on game-level videos perform better, likely due to the scene-dependent nature of sports activities.

For the shuttlecock dataset, performance is generally lower compared to the tennis ball dataset, likely because the tennis ball is larger, more uniformly textured, and rounder in shape. However, our motion-aware fusion mechanism still achieves improvements of over 0.8%, highlighting the importance of incorporating motion for enhanced tracking and prediction.

IV. CONCLUSION

In this paper, we present TrackNetV4, an advanced tracking framework that integrates motion-aware fusion to enhance the tracking and prediction of fast-moving, small objects in sports videos. TrackNetV4 builds on existing tracking technologies, incorporating a novel fusion mechanism that significantly improves performance in challenging conditions, such as occlusions and limited visibility. Our extensive experimental results highlight substantial performance improvements over the previous TrackNetV3, showcasing TrackNetV4's superiority in high-speed object tracking. Notably, TrackNetV4's lightweight and modular design allows for easy integration into various applications.

ACKNOWLEDGMENT

Arjun Raj conducted this research under the supervision of Lei Wang for his COMP3770 Computing Research Project (R&D) at ANU. He is a recipient of research sponsorship from Active Intelligence Australia Pty Ltd in Perth, Western Australia, including The Active Intelligence Research Challenge Award. This work was also supported by the NCI National AI Flagship Merit Allocation Scheme, and the National Computational Merit Allocation Scheme 2024 (NCMAS 2024), with computational resources provided by NCI Australia, an NCRIS-enabled capability supported by the Australian Government.

APPENDIX

PROJECT WEBSITE, CODE, AND MODEL

Our project website is here. You can find the code and pretrained models through the links provided on the site.

TRACKNET FAMILY OF MODEL ARCHITECTURES

Below, we provide details of the existing TrackNet frameworks and our TrackNetV4[2].

TrackNetV1 introduced in 2019

TrackNetV1 (Fig. 5), introduced by Huang *et al*. in 2019 [6], was the first deep learning framework specifically designed for tracking high-speed, small objects in sports applications. The architecture consists of two main components: (i) a VGG-16-based model for object classification, and (ii) a DeconvNet for semantic segmentation. To achieve pixel-level precision in predicting ball locations, the framework employs upsampling techniques to recover information lost in the max-pooling layers. The model uses a symmetric design, with an equal number of upsampling and max-pooling layers, ensuring balanced feature extraction and recovery during the tracking process.

TrackNetV2 introduced in 2020

TrackNetV2 [10] is an enhanced version of TrackNetV1, maintaining the same encoder-decoder architecture. The encoder, similar to that of TrackNetV1, leverages VGG-16 to generate feature maps by capturing image clues through convolutional kernels and condensing features via max-pooling operations. The decoder, structured symmetrically to the encoder's downsampling layers, performs upsampling to produce prediction heatmaps with the same resolution as the input images.

TrackNetV2 was designed to improve upon the first-generation model in several key areas, including processing speed, prediction accuracy, and GPU memory efficiency. Key enhancements include a multi-input, multi-output design that allows for more efficient handling of data, the incorporation of skip connections to facilitate better gradient flow and feature preservation, and the introduction of a new weighted Binary Cross-Entropy loss function, which improves performance, particularly in imbalanced data scenarios. These improvements

[2]We use TrackNetV2 as the backbone for the sake of simplicity.

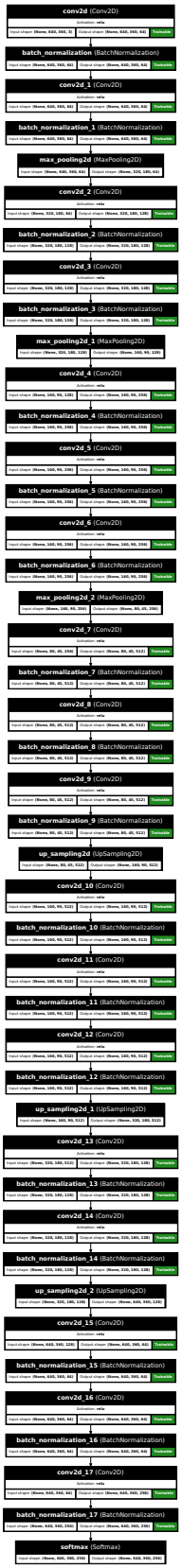


Fig. 5. TrackNetV1.

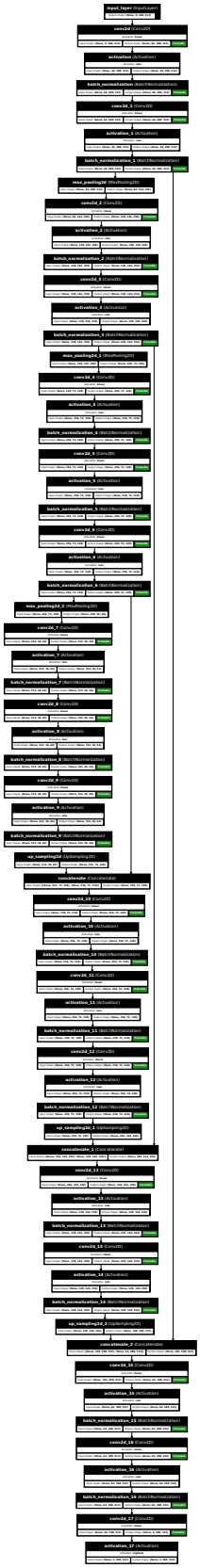


Fig. 6. TrackNetV2.

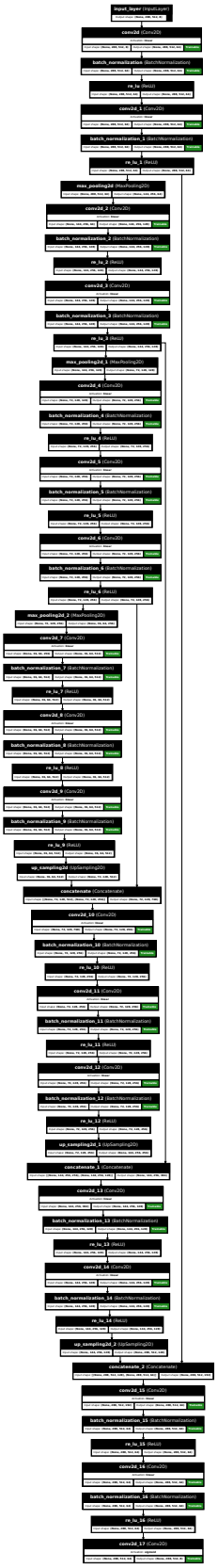


Fig. 7. TrackNetV3.

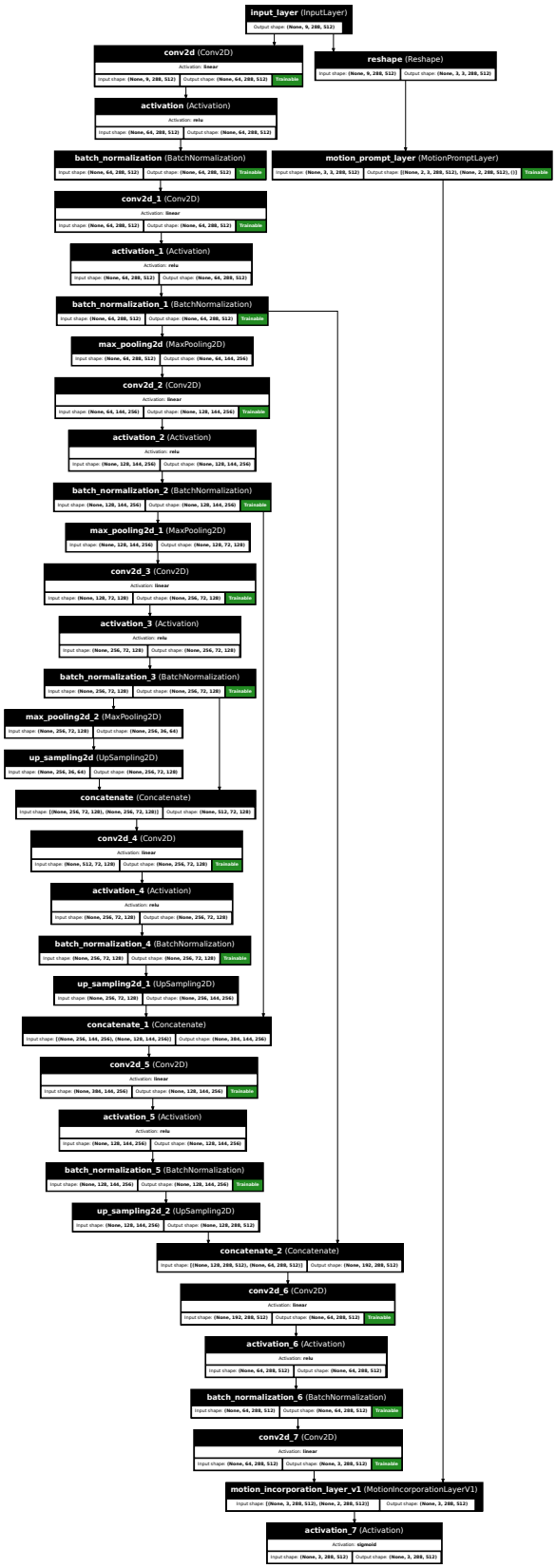


Fig. 8. Our TrackNetV4 model integrates modern motion concepts into the traditional 2D CNN-based TrackNet family (with TrackNetV2 chosen for visualization purposes). It enables precise, high-speed tracking of small objects in sports activities. We aim to inspire renewed interest in revisiting these older, yet still powerful models by enhancing them with contemporary motion concepts [1].

(a) Singles match. (b) Doubles match. (c) Visible balls in play on two courts.

(d) Multiple balls in play. (e) Nighttime match. (f) Ball camouflages with the court. (g) Balls blend with the court.

Fig. 9. Our multi-ball tracking dataset includes (a) singles and (b) doubles matches, with challenges such as (c) visible balls in play on two courts within a single video, (d) multiple balls in play, and scenarios that are challenging for ball tracking, including (e) nighttime matches, (f) balls camouflaged by the court's color, and (g) balls blending into the court's color. The dataset also features a range of resolutions.

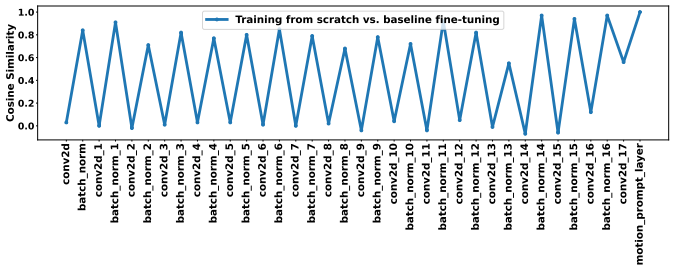


Fig. 10. Per-layer weight similarity measured using cosine similarity between training from scratch and baseline fine-tuning on TrackNetV4. Note that non-trainable layers are not included. We observe significant differences in the per-layer weights, particularly in the 2D convolutional layers, despite both models achieving very similar performance.

make TrackNetV2 more robust and efficient for high-speed, small-object tracking tasks in sports. Fig. 6 shows the model architecture of TrackNetV2.

TrackNetV3 introduced in 2023

TrackNetV3 [3] is a sophisticated model designed to improve the precision of ball localization in sports applications. It introduces two key modules: (i) Trajectory prediction, which builds on TrackNetV2 by considering a sequence of video frames to generate corresponding heatmaps that indicate the ball's position over time, and (ii) Trajectory rectification, which refines the predicted trajectory by generating repair masks to assess and correct errors, significantly enhancing the tracking accuracy and completeness of the trajectory.

In addition to these modules, TrackNetV3 uses an estimated background as supplementary data to more precisely locate the ball. Furthermore, inspired by the concept of image inpainting, the model defines an inpainting mask to identify frames that may require correction due to inaccuracies. The trajectory rectification module then takes this mask, along with the predicted trajectory from the tracking module, as inputs to produce a refined, corrected trajectory.

Since the primary focus of this work is the tracking framework, we visualize its architecture in Fig. 7.

Our TrackNetV4

Existing tracking frameworks primarily rely on visual features extracted by a VGG-16 network, which are then passed to a DeconvNet model acting as a decoder to predict the pixel-level location of the ball. However, for accurately tracking high-speed, small objects in sports videos, visual features alone are insufficient. These objects require motion information to ensure precise tracking and trajectory prediction.

Inspired by recent advancements in fine-grained video classification tasks [1], we propose explicitly incorporating motion concepts into the tracking framework. To achieve this, we introduce motion attention maps into the 2D CNN-based TrackNet family – an architecture that, while traditional, remains highly effective, *e.g.*, TrackNetV2. These motion attention maps are fused with high-level visual feature maps, enhancing motion-sensitive visual representations and improving tracking performance. Fig. 8 shows our model architecture.

Our approach aims to leverage and revitalize existing expert-designed model architectures by integrating modern motion concepts. Through this, we hope to spark renewed interest in these older but still powerful models by demonstrating their potential when combined with contemporary motion-based enhancements.

Challenging Multi-Ball Tracking Dataset

A significant issue with existing high-speed, tiny object tracking datasets is their simplicity. These datasets typically feature only a single moving ball on a clean court, leading to relatively straightforward scenarios.

(a) Attention maps from the tennis ball tracking dataset [6].

(b) Attention maps from the tennis ball tracking dataset [6].

(c) Attention maps from the shuttlecock tracking dataset [10].

(d) Attention maps from the shuttlecock tracking dataset [10].

(e) Attention maps from the multi-ball tracking dataset.

(f) Attention maps from the multi-ball tracking dataset.

Fig. 11. Motion attention map visualizations from the tennis ball tracking [6], shuttlecock tracking [10], and our multi-ball tracking datasets. We select the best model for each dataset to visualize the motion attention maps. Our motion attention maps effectively highlight movements, including those of small objects like balls.

To address this limitation, we collect our own dataset from online sources, incorporating more challenging scenarios, such as multiple moving balls and complex court layouts, which may include more than one court. The aim of our dataset is to demonstrate that TrackNetV4 excels at identifying the primary moving ball in multi-ball scenarios.

Fig. 9 presents visualizations of video frames from our dataset. Our dataset: (i) includes both single- and multi-ball scenarios, with all balls labeled and the primary ball highlighted, (ii) contains videos of varying resolutions, (iii) primarily features amateur games, and (iv) includes both singles and doubles matches. In total, we have collected over 23,000 training frames and more than 1,000 testing frames. Performance and speed metrics are reported.

Experiments and Discussion

We use TrackNetV2 as a baseline to demonstrate that, when enhanced with our motion attention maps and motion-aware fusion, referred to as TrackNetV4, it performs effectively on our challenging tennis ball tracking dataset. For all experiments, we adhere to the parameter settings specified by the original authors for training models from scratch, including

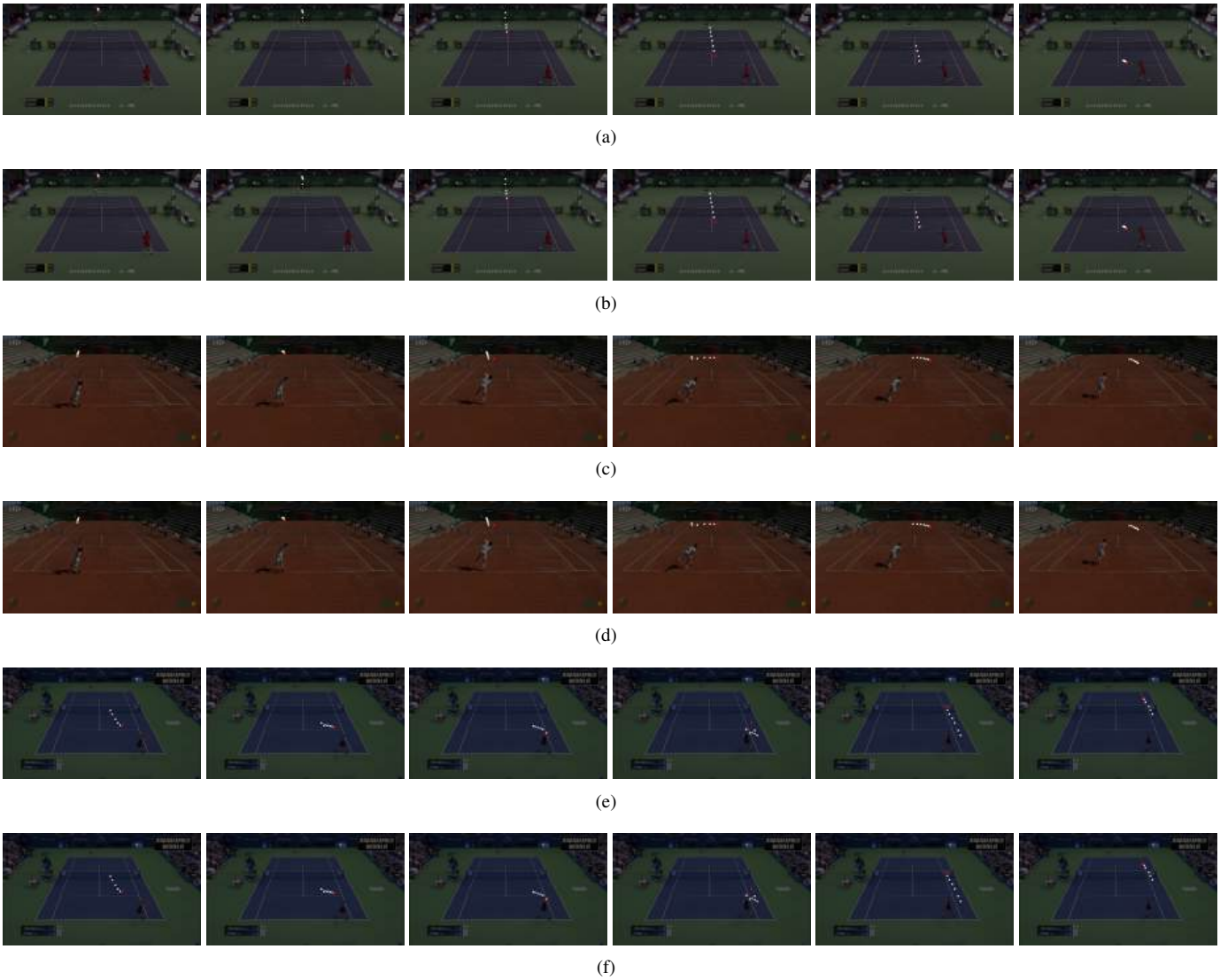


Fig. 12. Visualization of ball trajectories on the test set of tennis ball tracking dataset [6]. (a), (c) and (e) represent the baseline results, while (b), (d) and (f) show the results from our TrackNetV4. Best viewed with zoom for enhanced detail.

TABLE II

PERFORMANCE AND SPEED COMPARISONS BETWEEN THE BASELINE TRACKNETV2 AND TRACKNETV4 (THE BASELINE WITH OUR MOTION-AWARE FUSION) ON OUR DATASET ARE SHOWN. ROWS HIGHLIGHTED IN BLUE REPRESENT TRACKNETV4. VERSION 1 USES EQ. (4), WHILE VERSION 2 EMPLOYS THE MEAN MOTION ATTENTION MAP FOR ELEMENT-WISE MULTIPLICATION WITH EACH HIGH-LEVEL VISUAL FEATURE MAP. WE ALSO REPORT PERFORMANCE RESULTS FOR END-TO-END FINE-TUNING OF THE PRETRAINED TRACKNETV2. THE FINE-TUNING LEARNING RATES ARE SET TO $1e-3$, $1e-4$, AND $1e-5$, RESPECTIVELY, FROM TOP TO BOTTOM IN THE FINETUNING RESULTS.

	Method	**Total**	**Confusion matrix**					**Performance**				**Speed**
			TP	TN	FP1	FP2	FN	Acc.	Prec.	Rec.	F1	
Train from scratch	TrackNetV2	3279	2999	62	69	7	142	93.4	97.5	95.5	96.5	186.4
	TrackNetV4 (version 1)	3279	2971	40	76	29	163	91.8	96.6	94.8	95.7	174.7
	TrackNetV4 (version 2)	3279	3047	57	29	12	134	**94.7**	**98.7**	**95.8**	**97.2**	169.1
Fine-tuning	TrackNetV4 (version 1)	3279	2982	61	47	8	181	92.8	**98.2**	94.3	96.2	161.1
	TrackNetV4 (version 2)	3279	3050	60	44	9	116	**94.8**	**98.3**	**96.3**	**97.3**	168.5
	TrackNetV4 (version 1)	3279	3030	61	59	8	121	**94.3**	**97.8**	**96.2**	**97.0**	169.3
	TrackNetV4 (version 2)	3279	2721	63	35	6	454	84.9	**98.5**	85.7	91.7	171.3
	TrackNetV4 (version 1)	3279	2914	61	51	8	245	90.7	**98.0**	92.2	95.0	173.0
	TrackNetV4 (version 2)	3279	3022	54	94	15	94	**93.8**	96.5	**97.0**	**97.0**	179.2

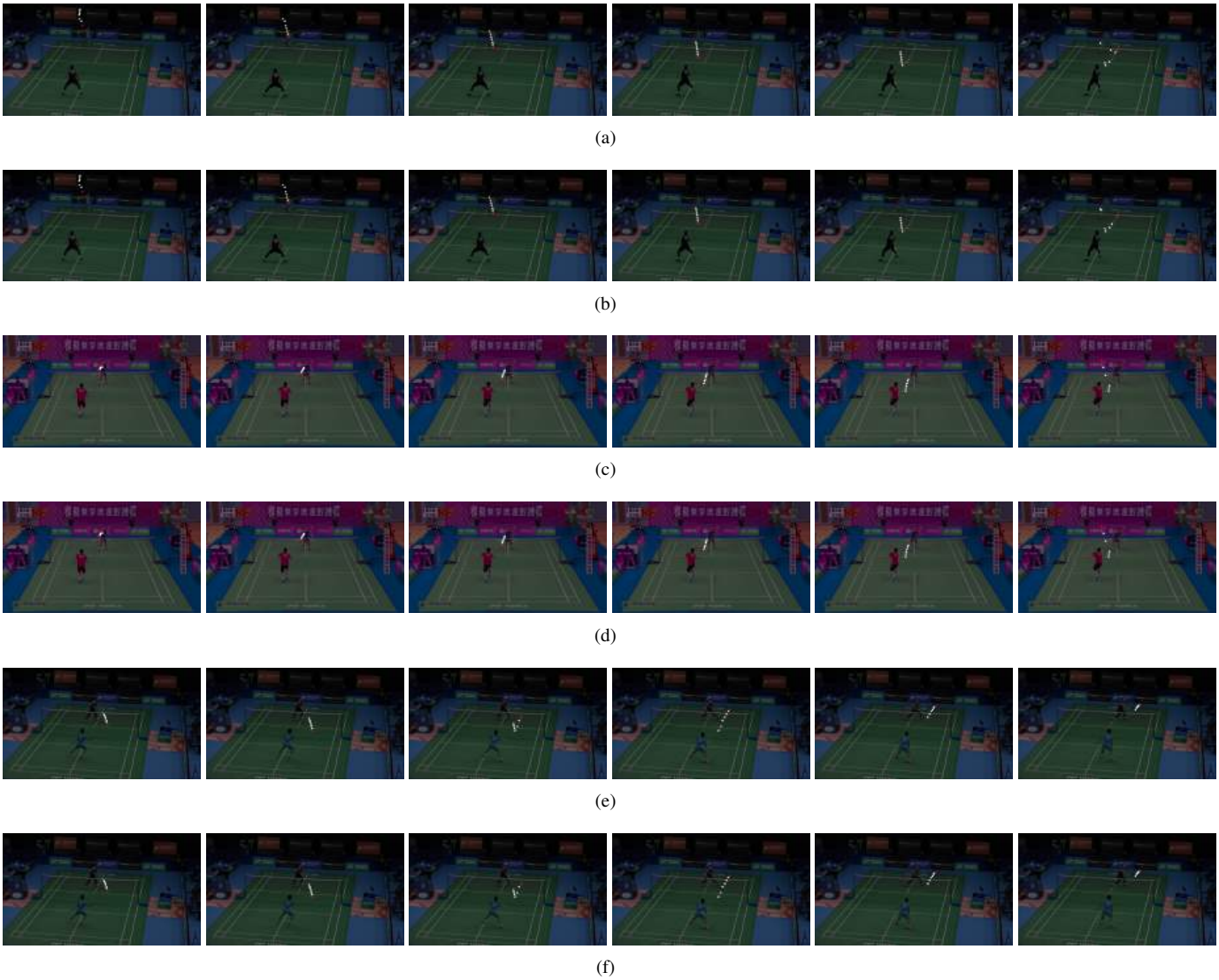


Fig. 13. Visualization of ball trajectories on the test set of shuttlecock dataset [10]. (a), (c) and (e) represent the baseline results, while (b), (d) and (f) show the results from our TrackNetV4. Best viewed with zoom for enhanced detail.

setting the number of training epochs to 30 and the initial learning rate to 1.0.

We select two fusion variants: version 1 follows Eq. (4), while version 2 uses the mean motion attention map for element-wise multiplication. We also present the results of end-to-end fine-tuning of TrackNetV4, starting from the pretrained model, using different learning rates. Table II summarizes the results.

As shown in the table, TrackNetV4 (version 2) trained from scratch outperforms the baseline, with improvements of 1.3%, 1.2%, 0.3%, and 0.7% in accuracy, precision, recall, and F1-score, respectively. Additionally, we observe that fine-tuning on top of the pretrained baseline further boosts performance, particularly in the recall metric.

The Role of Fine-Tuning

As shown in Table II, fine-tuning our TrackNetV4 using a pretrained TrackNetV2 model on our dataset generally improves performance, particularly for the version 2 model. For instance, with a learning rate of $1e{-}3$ and two additional learnable parameters (TrackNetV4 version 2) on top of the TrackNetV2 pretrained baseline, performance increased by 1.4%, 0.8%, 0.8%, and 0.8% for accuracy, precision, recall, and F1-score, respectively. Even with version 1 of our TrackNetV4, we observed improvements over the TrackNetV2 baseline of 0.9%, 0.3%, 0.7%, and 0.5% for accuracy, precision, recall, and F1-score, respectively. This demonstrates that both our motion attention maps and motion-aware fusion mechanism play crucial roles in enhancing the network's learning capacity for motion concepts.

On the other hand, training from scratch provides a similar boost across all four performance metrics. Fig. 10 shows the per-layer weight similarity for both training from scratch and fine-tuning the baseline using TrackNetV4. We observe significant differences in the per-layer weights between fine-tuning

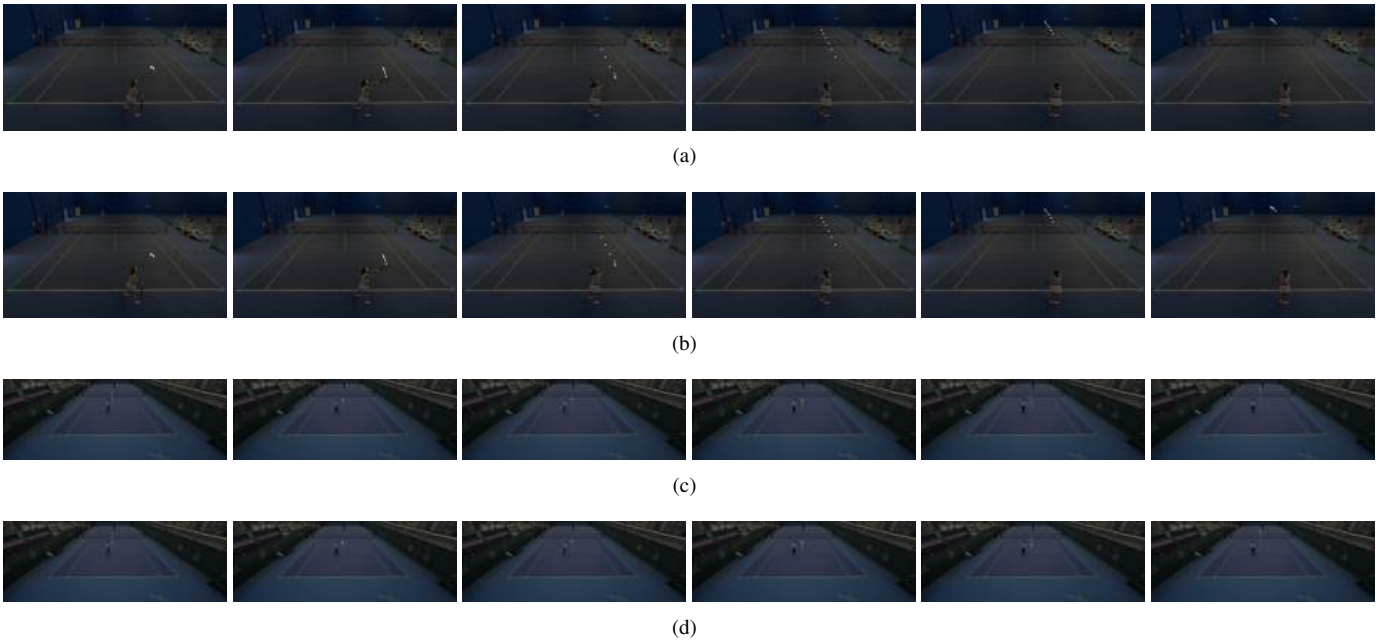


Fig. 14. Visualization of ball trajectories on the test set of our multi-ball tracking dataset. (a) and (c) represent the baseline results, while (b) and (d) show the results from our TrackNetV4. Best viewed with zoom for enhanced detail.

with the pretrained TrackNetV2 (using a smaller learning rate, such as $1e-3$) and training from scratch with TrackNetV4 (starting with a learning rate of 1.0), particularly in the 2D convolutional layers. We believe the motion prompt layer plays a crucial role in guiding the network to extract motion-aware features for tracking and prediction tasks. In the model trained from scratch, motion provides fresh information to the 2D CNN tracking framework. On the other hand, injecting motion attention maps into the pretrained model, even when using a fine-tuning strategy, still enhances tracking performance.

Visualizations of Motion Attention Maps

Fig. 11 shows visualizations of the learned motion attention maps generated by the best model for each dataset.

As illustrated, these motion attention maps effectively capture and highlight the motion dynamics within the videos, including small objects like balls. This ability to focus on fine details demonstrates the model's capacity to accurately represent subtle movements, enhancing its overall performance in high-speed and small-object tracking tasks.

Visualization of Ball Trajectories

Fig. 12, 13 and 14 present visualizations of predicted ball locations in the form of trajectories.

As shown in these figures, we observe that (i) the trajectories from both the baseline model (TrackNetV2) and our TrackNetV4, which builds on top of the baseline, are very similar for both the tennis ball and shuttlecock tracking datasets, and (ii) our TrackNetV4 on our multi-ball tracking dataset, using motion attention maps and motion-aware fusion, generates significantly smoother and more accurate ball trajectories. This highlights the effectiveness of incorporating motion concepts for accurately tracking and predicting fast-moving, small objects.

Additional Visualizations

Below in Fig 15, 16 and 17, we present additional visualizations comparing feature maps and heatmaps with and without our motion-aware fusion.

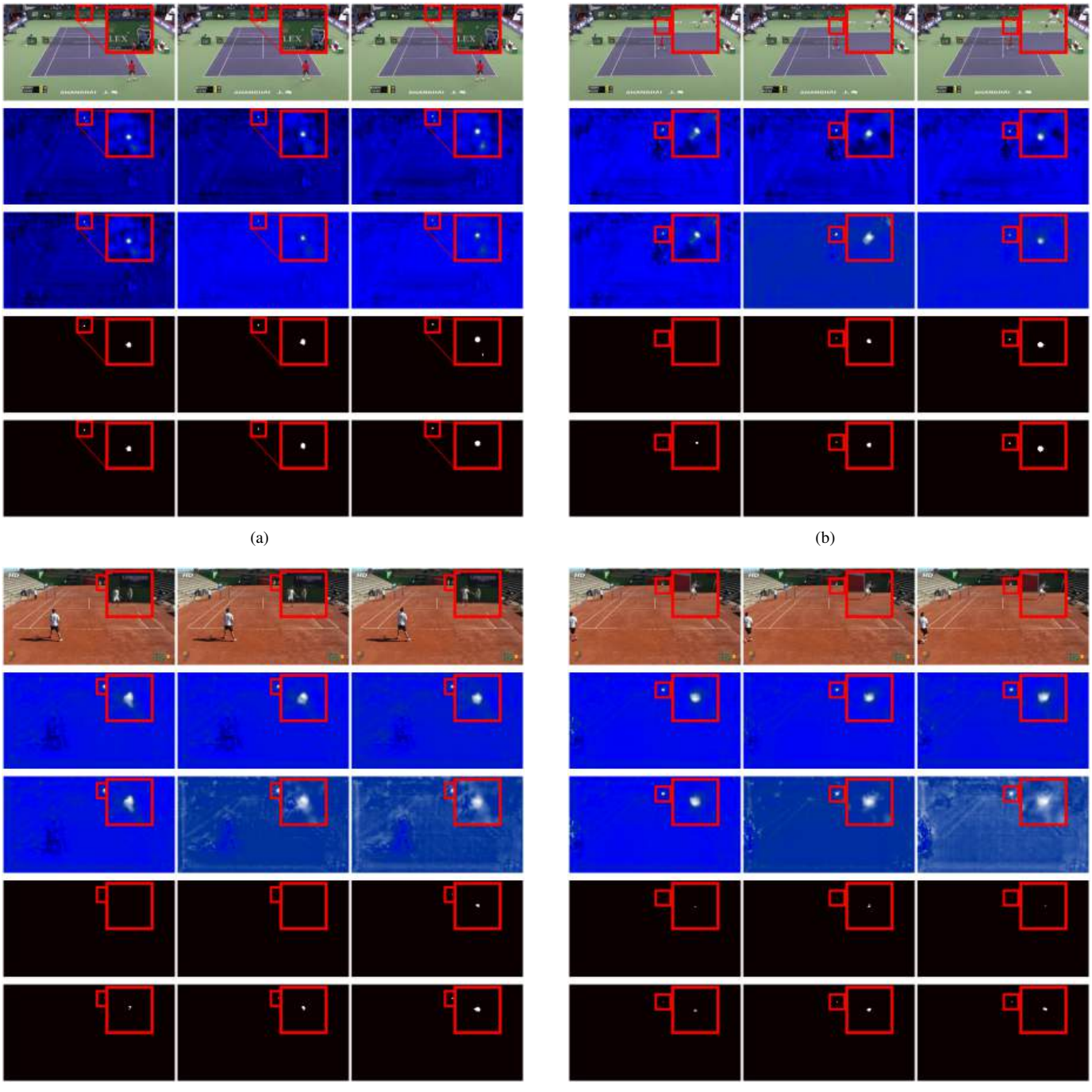
We observe that the motion-aware fusion significantly enhances the tracking and prediction of ball locations, as demonstrated by the clearer visualizations of both the feature maps (before applying the Sigmoid function) and the generated heatmaps. Notably, our feature maps show greater clarity, and the resulting heatmaps exhibit enhanced robustness, even when the balls are small, blurry, have afterimage trails, or are nearly invisible. This highlights the effectiveness of our method, which remains lightweight while successfully tracking high-speed, small objects in sports scenarios.

Questions and Answers

Q1: What motivates you for this research project?

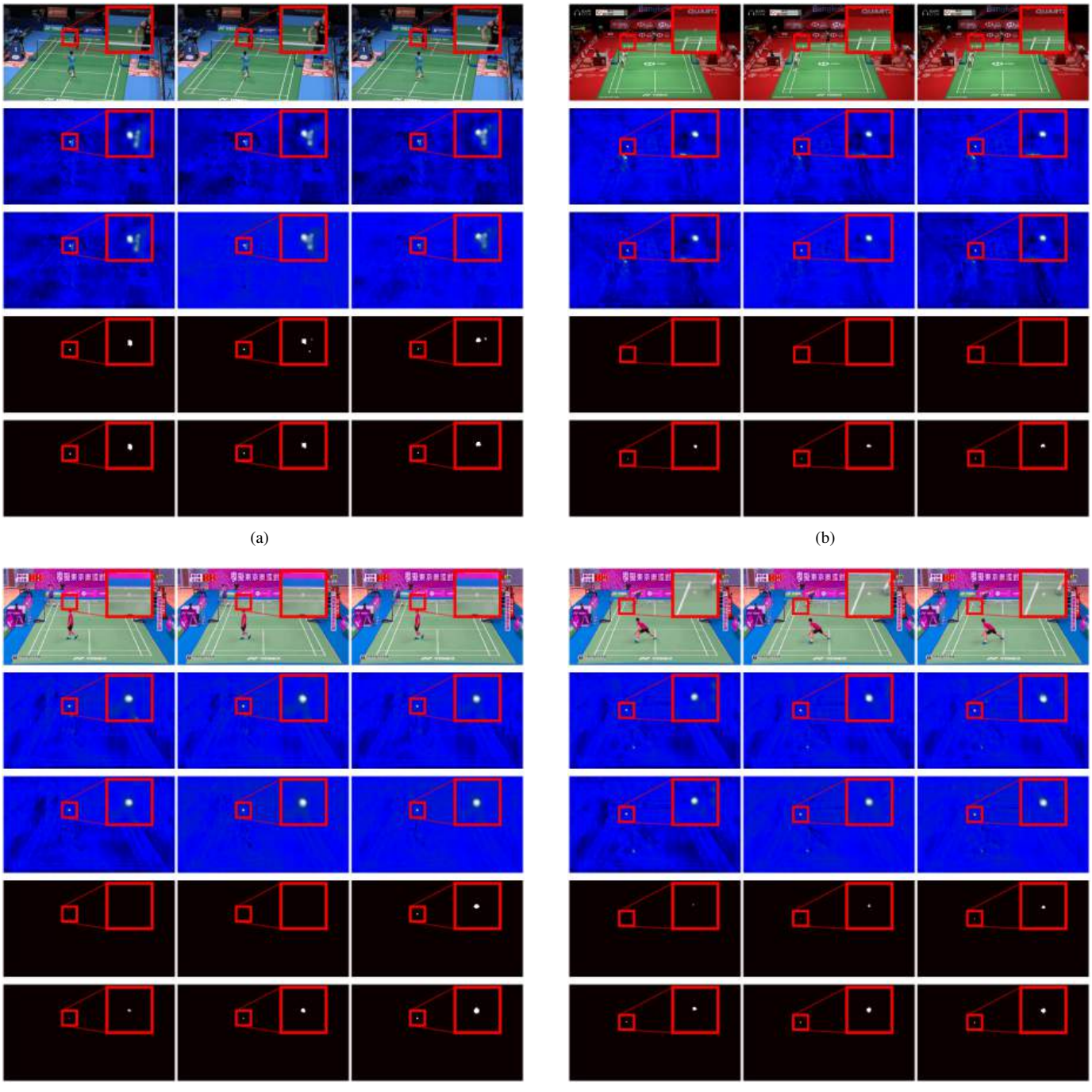
A1: AI has powered numerous real-world applications, including smart coaching apps for sports analysis and player performance monitoring. One of our key motivations is to improve tracking and prediction accuracy for fast-moving, small objects in sports videos. Traditional tracking methods often struggle with high-speed objects and frequent occlusions, leading to inaccurate predictions and poor performance. By enhancing tracking capabilities with motion-based concepts, we aim to provide more reliable tools for sports analysis and other applications requiring precise object tracking.

Current TrackNet models rely on traditional 2D CNNs for video-based ball tracking. While these models are effective,



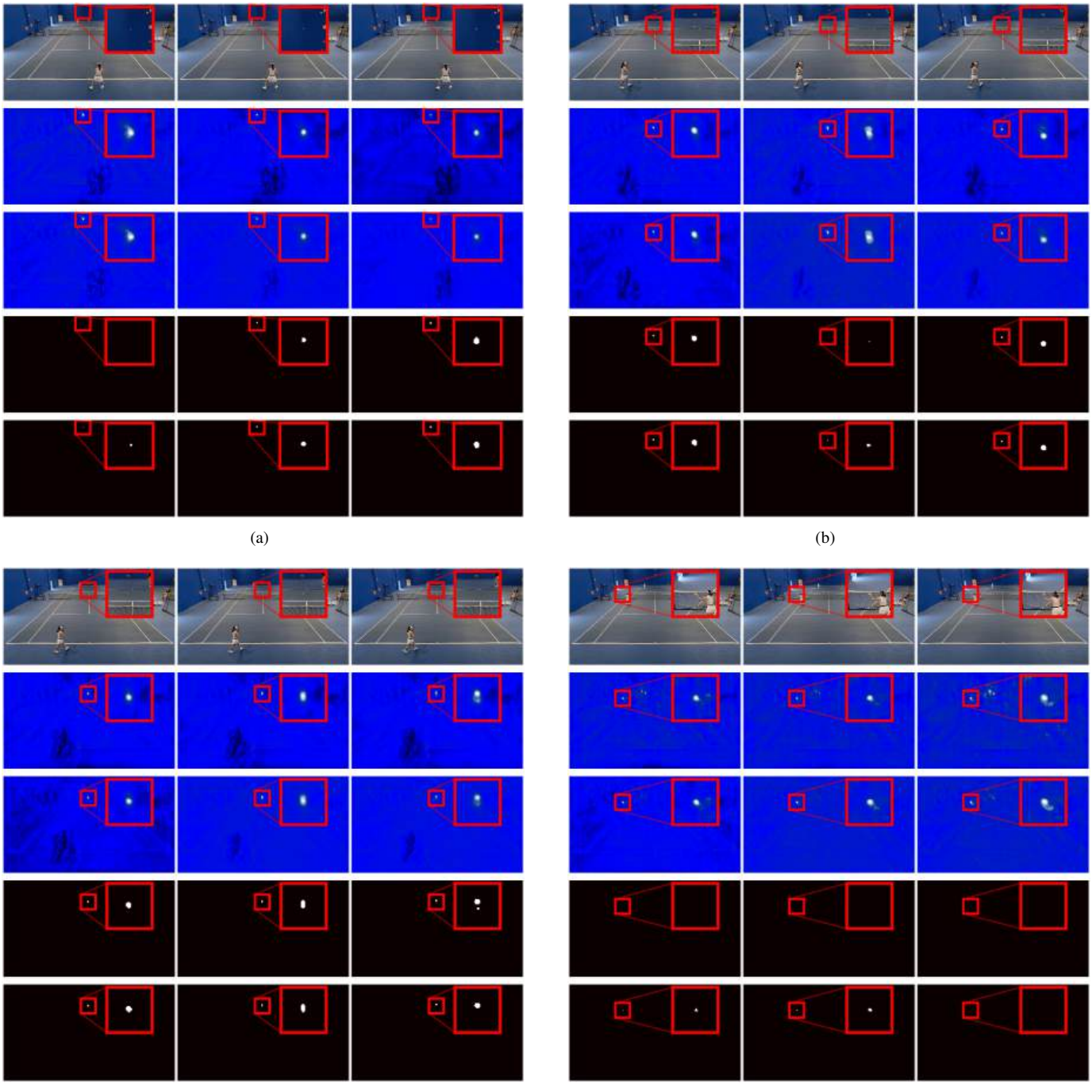
(a) (b) (c) (d)

Fig. 15. Comparison of feature maps and heatmaps with and without motion-aware fusion. We present four groups of visualizations from the tennis tracking dataset [6]. For each group, the first row displays the original video frame, the second and third rows show the feature maps from the baseline model and after applying motion-aware fusion, respectively. The fourth and fifth rows present the heatmaps from the baseline model and our TrackNetV4, respectively.



(a) (b) (c) (d)

Fig. 16. Comparison of feature maps and heatmaps with and without motion-aware fusion. We present four groups of visualizations from the shuttlecock dataset [10]. For each group, the first row displays the original video frame, the second and third rows show the feature maps from the baseline model and after applying motion-aware fusion, respectively. The fourth and fifth rows present the heatmaps from the baseline model and our TrackNetV4, respectively.



(a) (b)

(c)

Fig. 17. Comparison of feature maps and heatmaps with and without motion-aware fusion. We present four groups of visualizations from our multi-ball tracking dataset. For each group, the first row displays the original video frame, the second and third rows show the feature maps from the baseline model and after applying motion-aware fusion, respectively. The fourth and fifth rows present the heatmaps from the baseline model and our TrackNetV4, respectively.

there is certainly room for improvement. Our goal is to revisit these well-established, expert-designed architectures and enhance them by integrating modern motion prompts as guidance to boost ball tracking and prediction performance. We aim to reignite interest in reusing existing model architectures and pretrained models to reduce energy consumption and computing resource demands. We are the first to propose the concept of environment-friendly and reusable strategies in the fields of computer vision and machine learning.

Q2: What are the core innovations of TrackNetV4 and the reason for its name?

A2: TrackNetV4 builds on existing TrackNet models to enhance the tracking and prediction of high-speed, small objects in sports activities, using the latest motion attention maps and motion-aware fusion mechanisms. Using TrackNetV2 as a foundation, we introduce a motion prompt layer [1] that generates a sequence of motion attention maps. These maps are then fused with high-level visual feature maps through element-wise multiplication. Despite its simplicity, requiring only two additional learnable parameters, this fusion consistently improves accuracy, precision, recall, and F1-score.

We are the first to incorporate motion concepts into traditional 2D CNN-based tracking frameworks. TrackNetV4, when built on TrackNetV3 as the baseline architecture and augmented with motion attention maps and the fusion mechanism, outperforms TrackNetV3. These innovations lead to improved tracking performance and robustness, particularly in challenging scenarios with fast-moving objects and occlusions.

The core innovations of TrackNetV4 lie in its enhanced motion dynamics modeling, advanced fusion techniques, and improved feature extraction mechanisms. The name 'TrackNetV4' marks its evolution from earlier versions, representing the fourth iteration with substantial improvements and new capabilities. Furthermore, by reusing established tracking architectures and integrating modern modules, TrackNetV4 establishes itself as a next-generation framework with measurable improvements across all performance metrics.

Q3: Would the new challenging multi-ball object tracking dataset be released in the future?

A3: Yes, the release of a new challenging multi-ball object tracking dataset is planned for the future. This dataset aims to address current limitations by providing more complex and diverse scenarios for object tracking, which will facilitate the development and evaluation of more robust tracking algorithms.

Our project website includes visualizations of ball tracking performance on the test set of this dataset. Additionally, we provide access to our model code, best-performing models, and the inference, testing, and evaluation scripts for interested researchers.

Q4: How does TrackNetV4 advance object tracking and model reuse?

A4: TrackNetV4 has a range of potential applications, including real-time sports analysis, video surveillance, autonomous vehicles, and robotics. Its enhanced tracking capabilities make it suitable for scenarios requiring precise and reliable object tracking, even in dynamic and cluttered environments.

TrackNetV4 represents a pioneering step towards reusable models and environmentally-friendly training concepts. We aim to inspire renewed interest in integrating modern, innovative modules into existing expert-designed, pretrained models with minimal modifications. This approach, from a practical perspective, seeks to benefit the research community by enhancing the utility of established models.

REFERENCES

[1] Qixiang Chen, Lei Wang, Piotr Koniusz, and Tom Gedeon. Motion meets attention: Video motion prompts. *Asian Conference on Machine Learning (ACML)*, 2024.

[2] Wenshuo Chen, Hongru Xiao, Erhang Zhang, Lijie Hu, Lei Wang, Mengyuan Liu, and Chen Chen. Sato: Stable text-to-motion framework. *ACM Multimedia (ACM-MM)*, 2024.

[3] Yu-Jou Chen and Yu-Shuen Wang. Tracknetv3: Enhancing shuttlecock tracking with augmentations and trajectory rectification. In *Proceedings of the 5th ACM International Conference on Multimedia in Asia*, MMAsia '23, New York, NY, USA, 2024. Association for Computing Machinery.

[4] Haoyi Duan, Yan Xia, Zhou Mingze, Li Tang, Jieming Zhu, and Zhou Zhao. Cross-modal prompts: Adapting large pre-trained models for audio-visual downstream tasks. *Advances in Neural Information Processing Systems*, 36, 2024.

[5] Sen Fang, Lei Wang, Ce Zheng, Yapeng Tian, and Chen Chen. Signllm: Sign languages production large language models. *arXiv preprint arXiv:2405.10718*, 2024.

[6] Yu-Chuan Huang, I-No Liao, Ching-Hsuan Chen, Tsì-Uí İk, and Wen-Chih Peng. Tracknet: A deep learning network for tracking high-speed and tiny objects in sports applications. In *2019 16th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, pages 1–8. IEEE, 2019.

[7] Bing Li, Jiaxin Chen, Xiuguo Bao, and Di Huang. Compressed video prompt tuning. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, editors, *Advances in Neural Information Processing Systems*, volume 36, pages 31895–31907. Curran Associates, Inc., 2023.

[8] Hyeonwoo Noh, Seunghoon Hong, and Bohyung Han. Learning deconvolution network for semantic segmentation. In *Proceedings of the IEEE international conference on computer vision*, pages 1520–1528, 2015.

[9] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[10] Nien-En Sun, Yu-Ching Lin, Shao-Ping Chuang, Tzu-Han Hsu, Dung-Ru Yu, Ho-Yi Chung, and Tsì-Uí İk. Tracknetv2: Efficient shuttlecock tracking network. *2020 International Conference on Pervasive Artificial Intelligence (ICPAI)*, pages 86–91, 2020.

[11] Chien-Yao Wang, Alexey Bochkovskiy, and Hong-Yuan Mark Liao. Yolov7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 7464–7475, 2023.

[12] Hao Wang, Fang Liu, Licheng Jiao, Jiahao Wang, Zehua Hao, Shuo Li, Lingling Li, Puhua Chen, and Xu Liu. Vilt-clip: Video and language tuning clip with multimodal prompt learning and scenario-guided optimization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 5390–5400, 2024.

[13] Lei Wang and Piotr Koniusz. Flow dynamics correction for action recognition. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3795–3799. IEEE, 2024.

[14] Lei Wang, Ke Sun, and Piotr Koniusz. High-order tensor pooling with attention for action recognition. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3885–3889. IEEE, 2024.

[15] Lei Wang, Xiuyuan Yuan, Tom Gedeon, and Liang Zheng. Taylor videos for action recognition. *International Conference on Machine Learning (ICML)*, 2024.

[16] Jinglin Xu, Yijie Guo, and Yuxin Peng. Finepose: Fine-grained prompt-driven 3d human pose estimation via diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 561–570, 2024.

Arjun Raj is a Research Student at the School of Computing, Australian National University (ANU), supervised by Lei Wang. Previously, he worked as a Research Intern at Active Intelligence Australia Pty Ltd, Perth, where he received the 1st Annual Active Intelligence Research Challenge Award. At Lei's Temporal Intelligence and Motion Extraction (TIME) Lab, Arjun focuses on developing advanced techniques for tracking and detecting high-speed, small objects in sports analysis. His research interests include object detection and tracking, video processing, reusable models, efficient and cost-effective training methods, and promoting sustainability in AI. He aims to merge practical, resource-efficient approaches with cutting-edge AI advancements, contributing to more sustainable and scalable solutions in the field.

Lei Wang received his M.E. degree in Software Engineering from The University of Western Australia (UWA), Perth, in 2018, and his Ph.D. in Engineering and Computer Science from the Australian National University (ANU), Canberra, in 2023. He is currently a Research Fellow at the ANU School of Computing, where he leads a dynamic research team of master's and honours students in the Temporal Intelligence and Motion Extraction (TIME) Lab. He is also a Visiting Scientist with the Machine Learning Research Group at Data61/CSIRO (formerly NICTA). Previously, Lei was a Visiting Researcher at both the Department of Computer Science and Software Engineering at UWA and Data61/CSIRO. Since 2018, he has worked as a full-time Computer Vision Researcher at iCetana Pty Ltd., Perth, and since 2021, he has also served as a Computer Scientist at Active Intelligence Australia Pty Ltd., Perth. Lei has authored numerous first-author papers in prestigious venues, including CVPR, ICCV, ECCV, ACM MM, TPAMI, IJCV, and TIP. He received the Sang Uk Lee Best Student Paper Award at the Asian Conference on Computer Vision (ACCV) 2022. He currently serves as a Guest Editor for the MDPI open-access journal Electronics (Q2, h-index 83), and as an Area Chair for both the International Conference on Pattern Recognition (ICPR 2024) and ACM Multimedia 2024. His research interests include action recognition, anomaly detection, computer vision, and machine learning. Lei is an active member of IEEE and ACM as a Student Member.

Tom Gedeon received the B.Sc. (Hons.) and Ph.D. degrees from the University of Western Australia, Perth, WA, Australia. He holds the Optus Chair in AI and the Director of the Optus Centre for AI, Curtin University, Perth. Before this, he was a Professor of computer science and the former Deputy Dean of the College of Engineering and Computer Science, Australian National University, Canberra, ACT, Australia. He remains an Honorary Professor at ANU. He has over 400 publications. His main research interests include responsive and responsible AI and underpinned by edge computing efficient AI. His focus is on the development of automated systems for information extraction, from eye gaze and physiological data, as well as textual and other data, and for the synthesis of the extracted information into humanly useful information resources, primarily using neural/deep networks and fuzzy logic methods. Prof. Gedeon has run a number of international conferences. He is the former President of the Asia Pacific Neural Network Assembly and the Computing Research and Education Association of Australasia. He is currently a member of the Australian Research Council's Medical Research Advisory Group. He has been nominated for VC's awards for postgraduate supervision at three universities. He has been the General Chair of the International Conference on Neural Information Processing (ICONIP) three times. He is an Associate Editor of the IEEE Transactions on Fuzzy Systems and the Neural Networks (INNS/Elsevier).