

# Game State and Spatio-Temporal Action Detection in Soccer Using Graph Neural Networks and 3D Convolutional Networks

J r mie Ochin<sup>1,2</sup><sup>a</sup>, Guillaume Devineau<sup>2</sup><sup>b</sup>, Bogdan Stanculescu<sup>1</sup> and Sotiris Manitsaris<sup>1</sup>

<sup>1</sup>Centre for Robotics, MINES Paris - PSL, France

<sup>2</sup>Footovision, France

**Keywords:** Spatio-Temporal Action Detection, Video Action Recognition, Sport Video Understanding, 3D Convolutional Neural Networks, Graph Neural Networks, Soccer, Game Structure, Sports Analytics, Soccer Analytics.

**Abstract:** Soccer analytics rely on two data sources: the player positions on the pitch and the sequences of events they perform. With around 2000 ball events per game, their precise and exhaustive annotation based on a monocular video stream remains a tedious and costly manual task. While state-of-the-art spatio-temporal action detection methods show promise for automating this task, they lack contextual understanding of the game. Assuming professional players' behaviors are interdependent, we hypothesize that incorporating surrounding players' information such as positions, velocity and team membership can enhance purely visual predictions. We propose a spatio-temporal action detection approach that combines visual and game state information via Graph Neural Networks trained end-to-end with state-of-the-art 3D CNNs, demonstrating improved metrics through game state integration.

## 1 INTRODUCTION

Sports Analytics, the practice of collecting and interpreting data from past performances to inform future decisions, was pioneered in soccer by Royal Air Force Wing Commander Charles Reep, who meticulously analyzed over 600,000 passing moves beginning in the 1950s (Mandadapu, 2024). Since then, the demand for such statistics has experienced a strong development with the increased ability to collect, process, store, and analyze large quantities of data, using, among other things, cameras, GPS, computers, and algorithms (Jha et al., 2022). However, if recent advances in deep learning have facilitated the automation of player tracking and pitch location from monocular videos - a convenient alternative to the wearable -, the task of precise event annotation from soccer videos is still performed manually by trained operators (Cartas et al., 2022).

Indeed, sport video understanding remains a highly challenging domain that encompasses a wide variety of tasks and applications. This article spec-

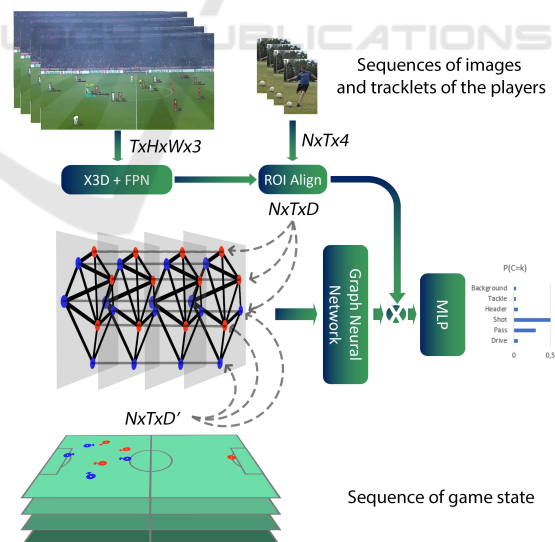




Figure 1: Spatio-temporal Action Detection in soccer, using a Graph Neural Network to encode Local Game State and a Track-Aware Action Detector (Singh et al., 2023) to extract relevant visual features. Our method demonstrates the complementarity of game state information and visual features and leads to improved performances.

<sup>a</sup> <https://orcid.org/0009-0008-3909-3757>

<sup>b</sup> <https://orcid.org/0009-0008-5202-0747>

ically addresses the spatio-temporal localization of events in soccer games. The goal is to accurately identify, in untrimmed videos, **when** an action occurs, **what** the action is, and **who** performed it, or **where** it took place since an action instance is defined as a set of linked bounding boxes over time, called action tube or tubelet, that contains the player performing the action. Solving this problem is essential for generating reliable and actionable statistics that can inform tactical decisions and allow opponent analysis or performance evaluations in professional soccer.

Current methods of spatio-temporal action detection are not designed to explicitly capture the complex relationships that can be observed in soccer gameplay. Soccer is a highly interactive sport, where player actions are decided based on the positions, movements of both teammates, opponents and their collective strategy. Existing methods, which rely on the aggregation of local visual features along the path of the players in screen space, lack the ability to fully comprehend these contextual relationships.

Jointly inspired by prior works in spatio-temporal action detection, group activity recognition and graph-based methods for soccer action spotting, our work proposes to enhance a track-aware spatio-temporal event detection method (TAAD) (Singh et al., 2023) by incorporating, for each player that is observed, contextual information about the local game state, through a learnt node embedding that is produced by a Graph Neural Network (GNN) (see Figure 1). We hypothesize that adding this information about the surrounding players, such as their positions and velocities on the pitch, team membership and appearance, structured by a GNN, can improve the performance of pure "visual" event detection models.

It is worth noting that using TAAD alongside game state information aligns naturally with soccer analytics, where tracking, identification, and position estimation are prerequisites for event annotation. Since tracklets and game state details are typically available before event annotation begins, it is natural to leverage them to enhance event detection.

We test our hypothesis by developing a new spatio-temporal action detection dataset dedicated to soccer, and by using a GNN trained end-to-end alongside a state-of-the-art 3D convolutional neural network (CNN), implementing a TAAD architecture. Given the applied nature of this research and the goal of enabling concrete applications, we aim to limit computational costs by ensuring that this implementation is trainable on a single entry-level professional GPU, such as an RTX A6000.

Our results demonstrate that explicitly incorporat-

ing structured game state information improves the performance of action detection models.

## 2 RELATED WORK

### 2.1 Spatio-Temporal Action Detection

In the field of video understanding, spatio-temporal action detection is the task of detecting intervals of possibly multiple and concurrent human actions in untrimmed videos. State-of-the-art methods are deep learning based and most of the work in this field addresses several core challenges:

- identify the actors,
- optionally model the relationship between them or with contextual elements,
- generate relevant features from the frames or video,
- sample and aggregate these features in a relevant manner,
- produce the frame-by-frame or clip-level action predictions,
- temporally link these predictions to obtain an action tube.

These methods are divided into two main categories: frame-level methods and clip-level methods.

**Frame-Level Methods:** the predictions are made frame by frame and then linked together to form action tubes. Conceptually, most of these methods rely on a region proposal network and ROI Pooling or ROI Align layers (He et al., 2017) to (i) detect the actors and (ii) locally sample and pool the features that are produced by a CNN. Early methods used 2D CNNs and optical flow for motion cues (Weinzaepfel et al., 2015; Saha et al., 2016; Singh et al., 2017), later transitioning to 3D CNNs for enhanced feature extraction (Gu et al., 2018; Girdhar et al., 2018). Expanding on this line of work, Track-Aware Action Detector (TAAD) first detects and tracks actors, using modern object detectors and tracking algorithms - which later facilitates the linking of predictions -, then aggregates features along the path for robustness to camera motion (Singh et al., 2023). Finally, some approaches integrate actor localization and action prediction, resembling YOLO's object detection approach (Redmon, 2016; Chen et al., 2021; Köpüklü et al., 2019).

Per-frame detections are linked using methods like dynamic programming with cost functions based on detection scores and overlaps (Saha et al., 2016; Singh et al., 2017; Kalogeiton et al., 2017) or tracking-by-detection (Weinzaepfel et al., 2015).

**Clip-Level Methods:** these methods predict tubelets, sequences of bounding boxes that tightly bind the actions of interest. The Action Tubelet Detector (ACT-detector) uses 2D CNNs to stack per-frame features for regressing the offset and dimensions of a cuboid anchor and predict the action class (Kalogeiton et al., 2017). Similarly, another approach estimates action centers and build cuboid anchors around them (Li et al., 2020). More recently, a Transformer-based method "encodes" video data using self-attention layers and "decodes" tubelets locations, classes, and scores using learned "tubelet queries" and cross-attention layers (Zhao et al., 2022).

**Visual Relationship Modeling:** the interaction between actors, objects, and their environment provide valuable context for improving detection and classification in videos. Several approaches have been proposed to model these relationships, such as attention mechanisms or GNNs.

Attention-based, the Actor Transformer Network (Girdhar et al., 2019) employs a region proposal network to sample features from a 3D CNN, creating actor representations. Then, the attention mechanism attend to all other spatio-temporal features extracted from the video, adding contextual information. Alternatively, attention can focus solely on actor tokens and a fixed-size environment representation obtained via average pooling of the feature map (Vo-Ho et al., 2021).

GNNs are also popular for modeling relationships, passing information between nodes representing entities and their relationships. Often, these methods first detect and track actors, sample features from 3D CNNs, and use these as node features, while edges connect actors, objects, or temporal sequences (Zhang et al., 2019; Tomei et al., 2021). Action predictions can occur on edges using embeddings of interacting entities or at the node level with entity-specific embeddings.

**Datasets:** most of the works described above were benchmarked on the UCF101-24 (Soomro et al., 2012) or AVA (Gu et al., 2018) datasets. They contain material that is substantially different from sports videos, where fast camera motions can happen, actors appear small on screen, with motion blur and, in team sports, with fluctuating visibility. Therefore, to foster the development of methods adapted to these characteristics, a dedicated multi-person video dataset of spatio-temporally localized sports actions was created: MultiSports (Li et al., 2021). It contains 3200 video clips collected from 4 sports (basketball, volleyball, soccer and aerobics gymnastics) and provides action tubelets annotations for these videos.

## 2.2 Group Activity Recognition in Sports

Group activity recognition consists in reasoning simultaneously over multiple actors in order to predict their individual and group activities. Some of the methods draw inspiration from visual relationship modeling, presented above. We will present only a few methods that tackle individual and group activity recognition in Volleyball.

In the early work Social Scene Understanding (Bagautdinov et al., 2017), a 2D CNN backbone is shared for player detection and activity recognition. A first stream produces candidate players detections, filtered using a Markov Random Field. Then, players features are sampled to produce fixed-size vector representations per frame. These are fed into an Recurrent Neural Network (RNN), whose hidden states enable individual action predictions frame-by-frame, while max pooling these states across players yields group activity predictions. Similarly, the Convolutional Relational Machine (Azar et al., 2019) uses 2D or 3D CNN features to generate "activity maps"—heatmaps indicating the likelihood of individual and group actions per class. Both approaches rely on the wide receptive field of CNNs for multi-agent reasoning, though this may lack the structural expressiveness offered by attention mechanisms or graphs.

Proposed more recently, the Actor-Transformer for Group Activity Recognition (Gavrilyuk et al., 2020) integrates RGB, Optical Flow, and Pose modalities. Players are "tokenized" via embedding features (using ROI Align) or pose coordinates, which serve as inputs to a Transformer encoder for individual and group action predictions. The centers of the bounding boxes in screen space are used as positional encoding of the players.

While both Visual Relationships Modeling methods and Group Activity Recognition methods effectively model relationships between entities within an image or video, they differ from approaches that leverage game state information, such as player positions, velocities, team membership and other contextual data, that are better suited to understand the complex contextual relationships that drive gameplay.

## 2.3 Graph Neural Networks in Sports

GNNs are ubiquitous. Their ability to model interactions within structured data has made them good candidates to model inter-agent patterns in a number of multi-agents problems. This section highlights methods applied to sports, particularly soccer, that use tabular player data (the "game state") as node representa-

tions rather than visual features, aiming to learn new discriminative features that could complement the visual ones.

In (Everett et al., 2023), a combination of GNN and LSTM is used to impute football players' locations at all time steps from sparse event data (e.g., shots, passes). Similarly, (Ding and Huang, 2020) employs a graph-attention network and dilated causal convolutions to predict trajectories of coordinated agents, such as basketball and football players. These studies highlight the strong link between players' spatial configurations and actions, as well as the ability of GNNs to deliver robust structured predictions.

Lastly, in (Cartas et al., 2022), a method for temporal event detection in football leverages player positions, velocities, and team identities (team 1, team 2, goalkeepers, referees). A GNN with edge convolutions and graph pooling encodes player configurations into single vectors for each time step, which are then used for action detection. Building on this approach, we propose a spatio-temporal action detection method that removes the graph pooling step to enable node-based predictions (per player predictions).

## 2.4 Vision-Language Models

Vision-Language Models (VLMs) integrate large language models with computer vision capabilities, enabling them to understand and generate text based on visual inputs. Recent notable examples include Molmo, Pixtral, LLaVa and Qwen2-VL (Deitke et al., 2024; Agrawal et al., 2024; Liu et al., 2024; Wang et al., 2024). However, VLMs are not yet mature enough for soccer video analysis because they rely heavily on proprietary models and synthetic data, lack specialized training datasets tailored to soccer, struggle to interpret complex player interactions and tactics, and tend to hallucinate - a critical flaw in the sensitive domain of sports analysis where accuracy is paramount.

# 3 METHODOLOGY

## 3.1 Dataset

The MultiSports dataset is a reference benchmark for multi-person spatio-temporal action detection methods in sports, but it is missing important features for this work: it lacks detection and tracking data of all the players as well as camera calibration models in order to estimate the positions of the players on the field. These are required features to study if game state

information can help enhancing pure visual spatio-temporal event detection methods. In order to remedy these issues, we created a specific "ball events" dataset that matches these requirements (see Figure 2). The **Footovision dataset** characteristics are:

- **20000 videos** of at least 75 frames (3 seconds).
- **997 different football games** from all over the world.
- **2500 samples per class** minimum.
- **8 classes of action:** ball-drive, pass, cross, header, throw-in, shot, tackle and ball-block

Sequences are sampled around randomly selected events from 997 games, ensuring no overlap between clips to create distinct training and validation sets. Some classes exceed 2,500 instances due to the high likelihood of certain events preceding or following others in a short period of time (e.g., ball-drives often lead to passes, crosses, or shots and are therefore present when a pass, cross or shot is sampled). This introduces a slight class imbalance.

The data used to make the annotations was collected from TV broadcast videos by the company Footovision, using in-house annotation software and team of professional annotators. Multiple manual quality controls are performed on tracking and identification data, line detection and camera calibration, and events are manually annotated with an estimated average precision of a few frames around the "ball touch". Ball-drives are annotated from the first ball touch to the next action (pass, cross, shot, tackle). All other events are annotated in a window of 7 frames around the ball touch. These annotations contain, frame by frame, for each video clip:

- the bounding box of the players and referees, with a tracklet ID to match bounding boxes across the clip (for the avoidance of doubt : this dataset does not contain the ball position),
- the team membership of the players (team 1 or 2),
- their shirt number,
- the class of the action performed by the player, including a background class when the players do not perform any action,
- their position and velocity in the pitch reference frame.

The videos in this dataset are subject to copyright protection and cannot be shared publicly.



		# classes	# inst.	Max #inst.	Min #inst.	# act. bbox	#non-act. bbox	Position & Velocity
<b>Multisports</b>	Aerobic gym.	21	8703	-	-	325k	0	X
<b>Multisports</b>	Volleyball	12	7645	-	-	139k	0	X
<b>Multisports</b>	Basketball	18	9099	-	-	213k	0	X
<b>Multisports</b>	Football	15	12254	-	-	225k	0	X
<b>Multisports</b>	Total	66	37701	3477	3	902k	0	X
<b>Footovision</b>	Football	8	19969	2500	2500	620k	19,9M	✓

Figure 2: Comparison of the characteristics of the MultiSports and Footovision datasets.

## 3.2 Visual Feature Extraction

### 3.2.1 Selection of a Baseline Method

We selected the Track-Aware Action Detector (TAAD) (Singh et al., 2023) as our baseline due to its suitability for sports with significant camera motion. TAAD tracks players and aggregates features from a state-of-the-art 3D CNN along the track using ROI Align (He et al., 2017), followed by a Temporal Convolutional Network (TCN) for per-frame action predictions. It performs well on the MultiSports dataset and aligns with this work, as player tracking is already completed before event annotation. We aim to compare TAAD with a hybrid approach combining visual features and local game state features learned by a GNN.

### 3.2.2 Players Visual Feature Extraction

The original TAAD implementation uses a SlowFast backbone (Feichtenhofer et al., 2019), a state-of-the-art CNN for video understanding. Nevertheless, recent works (Martin, 2020; Hong et al., 2022; Cioppa et al., 2024) demonstrate that lightweight architectures are particularly well-suited for small datasets of sports action recognition and tend to delay overfitting. For our implementation, we selected the X3D backbone family, which delivers performance comparable to SlowFast while using a fraction of the parameters. For instance, X3D-M achieves 76% top-1 accuracy with just 3.8M parameters, compared to SlowFast 4x16 R50, which achieves 75.6% top-1 accuracy with 34.4M parameters on the Kinetics-400 dataset (Feichtenhofer, 2020). This substitution reduces computational costs for subsequent end-to-end training as we plan to extend these architectures by using a GNN.

We utilize the first five blocks of the X3D backbone and integrate a Feature Pyramid Network (FPN) with the last three blocks, following (Lin et al., 2017; Singh et al., 2023). Player features are extracted using ROI Align based on their tracklet data, maintaining the feature dimension ( $D=192$ ) from X3D’s fifth

block. Additionally, this backbone preserves the temporal resolution, ensuring that a clip of  $T$  frames produces a feature tensor with a temporal dimension of  $T$ .

After applying TAAD to our clip of  $T$  frames, with  $N$  players, we obtain a feature tensor  $\Phi_{X3D} \in \mathbb{R}^{N \times T \times 192}$ , and we note  $\Phi_{X3D}^{(i,t)}$  the visual feature vector of the player  $i$  and time  $t$ :

$$\Phi_{X3D}^{(i,t)} = \text{ROI}(\text{X3D}(\text{sequence}), \text{bbox}^{(i,t)}) \quad (1)$$

Where  $\text{ROI}$  is the ROI Align sampling operation,  $\text{X3D}$  the application of the backbone on the sequence of images and  $\text{bbox}^{(i,t)}$  is the bounding box of player  $i$  at time  $t$ .

### 3.2.3 Trade-off Between Image Size, Length and Frame-Rate

Down-sampling the video resolution to 640x352 pixels provides a good balance for TAAD, as further reduction degrades performance, especially on wide fields of view where players appear small. No temporal sub-sampling was applied, as most player actions in the dataset are fine-grained and occur almost “instantly” (Hong et al., 2022). The 25 FPS rate aligns with the average event frequency (1 event every 65 frames) and accommodates the variability in event intervals: a ball drive can last for more than 10 seconds while a sequence of events such as shot and ball block can happen in a window of 10 frames or less. The clip length of 50 frames was chosen based on these resolutions and the available computational budget.

## 3.3 Local Game State Encoding

### 3.3.1 Game State Information

Our approach hypothesizes that incorporating structured information about players’ configuration on the pitch, referred to as the game state, complements purely visual methods like TAAD. In professional soccer, players’ actions, locations and velocities are

closely linked to the behavior of others, team membership, and collective strategies. Since inter-class variations in fine-grained action recognition are often subtle, modeling these player relationships can aid in learning additional discriminative features.

Moreover, we argue that using 2D positions and velocities on the pitch is a natural choice to model these inter-agents relations in a sport such as soccer. In contrast, methods for visual relationship modeling and group activity recognition rely on attention mechanisms that focus on visual features, lacking information about the spatial arrangement of the actors, or incorporating it only through screen space coordinates when using positional encoding. All the works that aim to model the game itself - estimate the probability of passes, next actions, detection of tactical patterns, trajectory forecasting... - are based on this very synthetic representation of the game state (Brefeld et al., 2019; Fadel et al., 2023; Martens et al., 2021; Rudolph and Brefeld, 2022; Dick et al., 2022; Anzer et al., 2022). In this paper, we adopt a hybrid approach, using both visual features and game state information:

We will call  $\mathbf{x}_i^t$  the feature vector of player  $i$  at time  $t$ , obtained by concatenating:

- its normalized position  $(p_x^{(i,t)}, p_y^{(i,t)})$  at time  $t$ ,
- its normalized velocity  $(v_x^{(i,t)}, v_y^{(i,t)})$  at time  $t$ ,
- its team membership, either 0 or 1,
- a projection of its visual feature vector  $\Phi_{X3D}^{(i,t)}$  in a space of lower dimension ( $D'=64$ ), noted  $\phi_{X3D}^{(i,t)}$ .

$$\mathbf{x}_i^t = \left( p_x^{(i,t)}, p_y^{(i,t)}, v_x^{(i,t)}, v_y^{(i,t)}, 0|1, \phi_{X3D}^{(i,t)} \right) \quad (2)$$

### 3.3.2 Graph of the Game

To be able to learn a node embedding meaningful for our problem, we build a spatio-temporal graph of the game  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$  following these steps :

- We create a node for each player at each time-step  $\mathcal{V} = \{\mathbf{x}_i^t, i \in [1, 22], t \in [1, T]\}$
- We create temporal edges between nodes of the same player at neighboring time-steps
- We create edges between nodes of different players at a given time-step using simple rules, such as the  $M$  closest players or using distance-based thresholds. In this implementation, we used the 6 closest players.

### 3.3.3 Edge Convolutions

A GNN can be described as a process by which the information contained in a graph evolves using functions approximated by neural networks. More precisely, it is the process by which the information contained in a node of that graph is modified by the information contained in the nodes of its neighborhood: vector messages are exchanged between connected nodes and updated using neural networks. This defining feature of GNN is called *neural message passing*.

In a GNN, layers of message passing are performed, and at each iteration  $k$ , the *hidden embedding*  $h_u^{(k)}$  of a node  $u \in \mathcal{V}$  is updated according to information *aggregated* from  $u$ 's graph neighborhood  $\mathcal{N}(u)$ .

Among the large variety of message passing methods, we chose Edge Convolutions with an asymmetric edge function to implement our GNN (Wang et al., 2019). Its update method can be summarized by equation 3:

$$h_u^{(k+1)} = \text{MAX} \left( \text{MLP} \left( h_u^{(k)} \parallel h_v^{(k)} - h_u^{(k)} \right) \right) \quad (3)$$

Where *MLP* is a shallow *Multi-layer Perceptron* and *MAX* a *channel-wise max pooling operation*.

This choice is driven by our will to build a node embedding that explicitly combines global information about the node, captured by  $h_u^{(k)}$ , with relative and local neighborhood information, captured by  $h_v^{(k)} - h_u^{(k)}$ . This is the reason why we call this node embedding the "local game state encoding".

Its benefit is particularly salient on the first layer of convolution ( $k=0$ ) where  $h_u^0 = \mathbf{x}_i^t$ : the edge convolution starts learning an embedding based on the absolute position and velocity information of the node, as well as the relative position and speed of its neighbors, or their team membership and variation of appearance. This is quite relevant given our hypothesis of high inter-dependency of motions between agents in football. Then, layer by layer, the network progressively learn a more abstract and more discriminative node embedding, simply guided by the loss function during training.

We only use a few layers of edge convolutions: in practice 3 or 4, to avoid the "over-smoothing" effect of the nodes.

## 3.4 Final Prediction

Finally, we concatenate  $\Phi_{X3D} \in \mathbb{R}^{N \times T \times D}$  with the tensor of nodes features obtained after application of the last layer  $K$  of graph edge convolution  $h^{(K)} \in \mathbb{R}^{N \times T \times D'}$ , apply the temporal convolution network of the TAAD

and produce a dense tensor of predictions using a using a Multi-layer Perceptron (MLP). We train our network in a supervised manner using a cross entropy loss function (CE-Loss).

### 3.5 Action Tube Construction

An event is defined by its list of characteristics: (start frame, end frame, class, score). Both TAAD and our method output dense predictions (class prediction per frame and per player), that can contain spurious detections on a few frames in the middle of a sequence of the same label.

Therefore, in order to transform this sequence into an action-tube we filter it by performing a label smoothing optimization (Saha et al., 2016; Singh et al., 2017; Kalogeiton et al., 2017). Only then the sequence of events is extracted and matched with the ground truth labels to calculate our metrics. A predicted event is added to the list of events when its class is not background.

### 3.6 Implementation Details

We used 50 frames as input with no sub-sampling (2 seconds of video clip). We used the X3D S pre-trained backbone as feature extractor and the PyTorch Geometric framework was used for the Edge Convolutions layers. We used the Adam optimizer (Kingma and Ba, 2015), with a learning rate of 0.0005, a weight decay of  $1e-05$  on the non-bias parameters, a gradient accumulation over 20 iterations before weight update and a batch size of 6 on a RTX A6000 GPU. We trained the system end-to-end for a total of 13 epochs and divided the learning rate by 10 at epoch 10.

## 4 EVALUATION

### 4.1 Metrics

We have chosen to report the mean Average Precision as our main metric. It is common in object detection and video action detection works (Weinzaepfel et al., 2015; Kalogeiton et al., 2017). A detection is correct when its Intersection-Over-Union (IoU) with a ground truth action tube is larger than a given threshold (e.g. 0.5 or 0.2) and the predicted label matches the ground truth one. Then, Average Precision is computed for each class, using the 11 points method as proposed by (Everingham et al., 2010), and the mean is computed across classes. The particularity of our setup is that the tracking of the players is already done: therefore, the IoU is purely temporal.

In more details, the matching between predicted and ground truth events is performed as follows: the temporal IoU is calculated for all event pairs to identify the best match, regardless of class. If the highest IoU exceeds the threshold, the events are matched and removed from their lists. Matches with identical classes are True Positives. Unmatched true events (he has not been detected) are paired with "dummy" predicted events of the background class, and unmatched predicted events are paired with "dummy" true events of the background class.

### 4.2 Comparisons

We implemented TAAD and TAAD + GNN as described in the Methodology section.

**Firstly**, we observe that metrics at 0.2 IoU threshold obtained when running our implementation of TAAD on the Footovision dataset are close to the metrics reported by the authors on their implementation of TAAD on the MultiSports dataset. Though these results cannot be compared directly (there can be slight domain gaps, the backbones are not identical, etc.), this shows that this method translates well to this new dataset, even with a lighter backbone:

- *60.6% mAP@0.2 and 37% mAP@0.5 on MultiSports*
- *59.58% mAP@0.2 and 46.4% mAP@0.5 on the Footovision dataset*

The gap observed on the mAP at a 0.5 threshold of IoU could be explained by the fact that our dataset is less challenging than MultiSports: less classes, less class imbalance, more samples per class on the minority classes.

**Secondly**, Figure 3 and Figure 4 show the higher performance of TAAD + GNN compared to TAAD alone. This demonstrates that integrating structured game state information into the detection process leads to improvements in detections.

Method	mAP	
	0,2	0,5
<b>TAAD</b>		
(X3D + FPN + TCN)	59,58	46,4
<b>TAAD + GNN (ours)</b>		
(X3D + FPN + Local Game State GNN)	<b>64,48</b>	<b>49,7</b>

Figure 3: Comparison of TAAD and our method TAAD + GNN: integrating structured game state information into the detection process leads to improvements in performances.

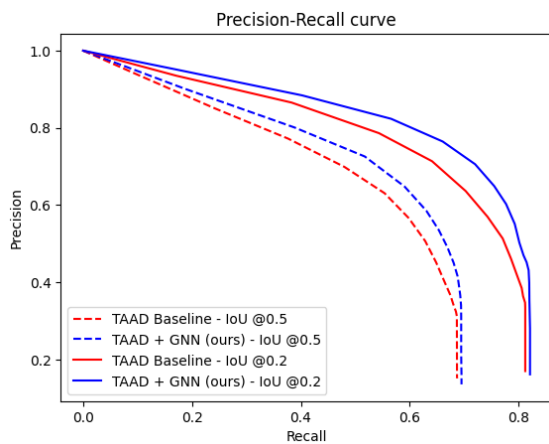


Figure 4: Comparison of the Precision - Recall curves for the TAAD and TAAD + GNN, with various IoU thresholds (0.2 and 0.5).

### 4.3 Discussion

#### 4.3.1 High Recall and Low Precision Regime

In order to meet the expectation of exhaustiveness of events detection in the context of data analytics, it is envisaged to work in a regime of high recall - low precision, using predictions with low confidence scores. Though this setting unavoidably leads to the detection of numerous False Positives, their subsequent filtering by human intervention remains far more competitive compared to the alternative: seeking missing information by "scrubbing" a video.

We assess the interest of our method in this context of high recall - low precision by studying the variations in the number of True Positives and False Positives when switching from TAAD to TAAD + GNN. For that purpose, we fixed the IoU threshold at 0.2 and the confidence score threshold at 0.5 (for an overall recall reaching ~ 80%).

The results are reported in Figure 5 and allow us to confirm the benefit of our method in this context: making more structured predictions by using game state information reduces the number of false positives by almost 30%, while keeping the overall number of True Positives constant.

#### 4.3.2 Detailed Variations of Performances

A more detailed analysis of the results shows that despite an overall improvement in the metrics, the situation is quite contrasted on a per class level.

Indeed, analyzing the class level results reported in Figure 5, we observe an important erosion of detection for the class "Cross" (almost -8%). It is partly explained by the higher confusion between "Cross" ac-

Actions	Increase in number of True Positives	Decrease in number of False Positives
Ball-Drive	0,0%	23,0%
Pass	-0,5%	13,1%
Cross	-7,9%	39,9%
Throw	-0,4%	31,0%
Shot	3,9%	23,3%
Header	0,9%	29,0%
Tackle	13,8%	47,8%
Ball-Block	1,3%	36,0%
Overall	0,4%	29,0%

Figure 5: When working in high recall - low precision regime, our method significantly reduce the number of False Positive detections.

tions and "Shot" actions when switching from TAAD to TAAD + GNN, while, surprisingly, less "Shots" are confused with "Crosses" (see the confusion matrices in Figure 6). These two classes can be easily confused: the players performing both actions are in the same zones of the pitch and both shoot the ball toward the goalmouth, except that in the first case the player intend to pass the ball to a player that is closer to the goalmouth, while in the second case he intends to score directly. It is also sometimes difficult to know what was the intention of the player when he misses one of those actions.

"Cross" is also the class that experience the second most important decrease in its number of False Positives. Therefore, a shift in the equilibrium recall-precision for the current confidence score threshold (0.5), between the two methods, could be an another factor of explanation.

Finally, the "Tackle class" experiences the most important improvement in both decrease of False Positives (almost 48%) and increase in True Positives (almost 14%). The game state information seems, on one hand, to limit the class confusions between "Tackle" and "Ball-drive" (around -52%) and, on the other hand, reduce the level of False Positives, as seen in the confusion matrices in Figure 6.

#### 4.3.3 The Benefits of Structured Predictions

In order to better understand in which cases structured prediction is beneficial, and explain the observed improvement in our metrics, we compiled, per class, the clips that contained false positive events when using the TAAD, but that didn't fool the TAAD + GNN approach.

One key benefit is the reduction of simultaneous false positives. TAAD occasionally detects actions that overlap with ongoing real actions, which is unrealistic given there is only one ball and one ac-



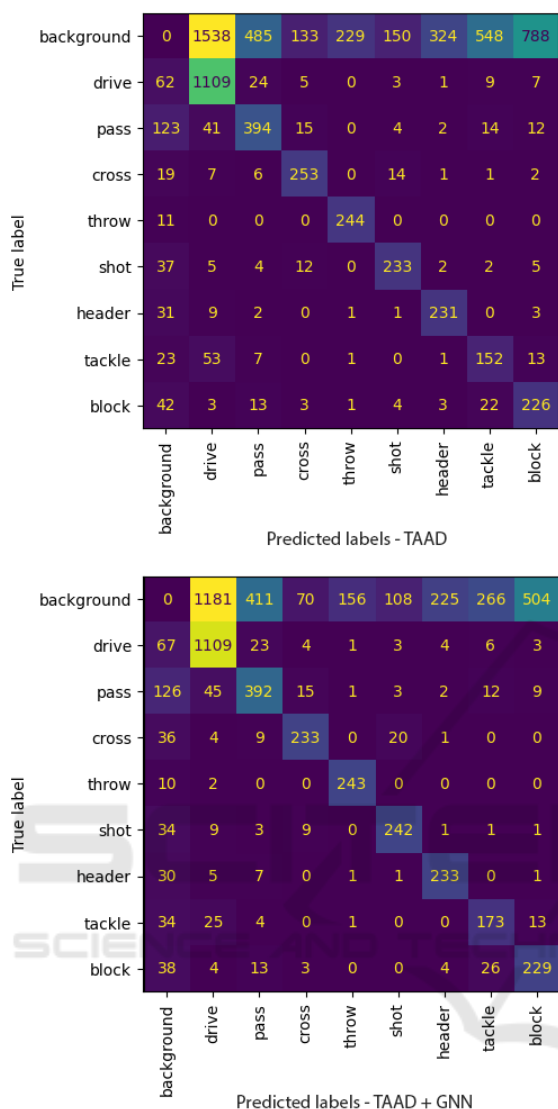


Figure 6: Confusion matrices obtained when applying TAAD and TAAD + GNN, with IoU threshold at 0.2 and confidence score threshold at 0.5.

tive player controlling it. For instance, TAAD might falsely detect a player on the sideline as performing a throw-in while another player is actively driving the ball, due to reliance solely on player appearance and gestures. By incorporating surrounding player information and their relative configurations, such false positives are eliminated, particularly in the Throw class. While filtering detections by score could be an alternative, 3D CNNs often exhibit overconfidence, with false positives sometimes scoring higher than the true action.

Another benefit is the reduction of false positives in actions like Ball-blocks or Header. False positives often occur when a player attempts the gesture

but misses the ball, where motion blur and camera perspective falsely suggest contact. These actions usually happen in crowded areas, and successful execution typically causes a noticeable ball trajectory change, prompting surrounding players to accelerate. When the ball is missed, the players’ movement remains less disrupted. Our method appears to compensate for 3D CNNs’ difficulty in linking ball trajectory to player contact by learning how groups of players react to sudden ball trajectory changes.

#### 4.3.4 Team Sports and Occlusion

Many false positives happen when players are really close to each others, leading either to ambiguous situations, when two players are equidistant from the ball in screen space, or even to impossible detections when players are not visible. Sometimes, two players can have their bounding boxes with very high overlap, which means that their features, extracted using a ROI Align layer, are almost identical. This is for example the case in many Header events when two players are performing aerial duels. For those hard cases, further research will be necessary, and it is highly probable that each class of event will require its own approach.

## 5 CONCLUSIONS

We propose a method that combines visual features and structured game state information, using Graph Neural Networks, to enhance more traditional spatio-temporal action detection methods in soccer. We demonstrate the benefit of our method in the context of high recall - low precision regime, required to generate reliable statistics more efficiently. Nevertheless, our method sees little of the game: only 2 seconds. In future works we envisage to push our hypothesis further and overcome this temporal horizon by (i) using our method to make “prior” predictions of actions for each player, (ii) use sequence to sequence methods to learn the logic and dynamic of the game on long sequences of events and leverage that knowledge to either generate “posterior” predictions per player, or denoise or score our sequences of detections, conditionally to the game state, and (iii) introduce additional game state features, traditionally found in the literature about game state based methods to predict future events.

## REFERENCES

- Agrawal, P., Antoniak, S., Hanna, E. B., Bout, B., Chaplot, D., Chudnovsky, J., Costa, D., De Monicault, B., Garg, S., Gervet, T., et al. (2024). Pixtral 12b. *arXiv preprint arXiv:2410.07073*.
- Anzer, G., Bauer, P., Brefeld, U., and Faßmeyer, D. (2022). Detection of tactical patterns using semi-supervised graph neural networks. In *16th MIT Sloan Sports Analytics Conference*.
- Azar, S. M., Atigh, M. G., Nickabadi, A., and Alahi, A. (2019). Convolutional relational machine for group activity recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7892–7901.
- Bagautdinov, T., Alahi, A., Fleuret, F., Fua, P., and Savarese, S. (2017). Social scene understanding: End-to-end multi-person action localization and collective activity recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4315–4324.
- Brefeld, U., Lasek, J., and Mair, S. (2019). Probabilistic movement models and zones of control. *Machine Learning*, 108(1):127–147.
- Cartas, A., Ballester, C., and Haro, G. (2022). A graph-based method for soccer action spotting using unsupervised player classification. In *Proceedings of the 5th International ACM Workshop on Multimedia Content Analysis in Sports*, pages 93–102.
- Chen, S., Sun, P., Xie, E., Ge, C., Wu, J., Ma, L., Shen, J., and Luo, P. (2021). Watch only once: An end-to-end video action detection framework. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8178–8187.
- Cioppa, A., Giancola, S., Somers, V., Magera, F., Zhou, X., Mkhallati, H., Delière, A., Held, J., Hinojosa, C., Mansourian, A. M., et al. (2024). SoccerNet 2023 challenges results. *Sports Engineering*, 27(2):24.
- Deitke, M., Clark, C., Lee, S., Tripathi, R., Yang, Y., Park, J. S., Salehi, M., Muennighoff, N., Lo, K., Soldaini, L., et al. (2024). Molmo and pixmo: Open weights and open data for state-of-the-art multimodal models. *arXiv preprint arXiv:2409.17146*.
- Dick, U., Link, D., and Brefeld, U. (2022). Who can receive the pass? a computational model for quantifying availability in soccer. *Data mining and knowledge discovery*, 36(3):987–1014.
- Ding, D. and Huang, H. H. (2020). A graph attention based approach for trajectory prediction in multi-agent sports games. *arXiv preprint arXiv:2012.10531*.
- Everett, G., Beal, R. J., Matthews, T., Early, J., Norman, T. J., and Ramchurn, S. D. (2023). Inferring player location in sports matches: Multi-agent spatial imputation from limited observations. *arXiv preprint arXiv:2302.06569*.
- Everingham, M., Van Gool, L., Williams, C. K., Winn, J., and Zisserman, A. (2010). The pascal visual object classes (voc) challenge. *International journal of computer vision*, 88:303–338.
- Fadel, S. G., Mair, S., da Silva Torres, R., and Brefeld, U. (2023). Contextual movement models based on normalizing flows. *AStA Advances in Statistical Analysis*, 107(1):51–72.
- Feichtenhofer, C. (2020). X3d: Expanding architectures for efficient video recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 203–213.
- Feichtenhofer, C., Fan, H., Malik, J., and He, K. (2019). Slowfast networks for video recognition. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 6202–6211.
- Gavrilyuk, K., Sanford, R., Javan, M., and Snoek, C. G. (2020). Actor-transformers for group activity recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 839–848.
- Girdhar, R., Carreira, J., Doersch, C., and Zisserman, A. (2018). A better baseline for AVA. *arXiv preprint arXiv:1807.10066*.
- Girdhar, R., Carreira, J., Doersch, C., and Zisserman, A. (2019). Video action transformer network. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 244–253.
- Gu, C., Sun, C., Ross, D. A., Vondrick, C., Pantofaru, C., Li, Y., Vijayanarasimhan, S., Toderici, G., Ricco, S., Sukthankar, R., et al. (2018). Ava: A video dataset of spatio-temporally localized atomic visual actions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6047–6056.
- He, K., Gkioxari, G., Dollár, P., and Girshick, R. (2017). Mask r-CNN. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969.
- Hong, J., Zhang, H., Gharbi, M., Fisher, M., and Fatahalian, K. (2022). Spotting temporally precise, fine-grained events in video. In *European Conference on Computer Vision*, pages 33–51. Springer.
- Jha, D., Rauniar, A., Johansen, H. D., Johansen, D., Riegler, M. A., Halvorsen, P., and Bagci, U. (2022). Video analytics in elite soccer: A distributed computing perspective. In *2022 IEEE 12th Sensor Array and Multichannel Signal Processing Workshop (SAM)*, pages 221–225. IEEE.
- Kalogeiton, V., Weinzaepfel, P., Ferrari, V., and Schmid, C. (2017). Action tubelet detector for spatio-temporal action localization. In *Proceedings of the IEEE international conference on computer vision*, pages 4405–4413.
- Kingma, D. and Ba, J. (2015). Adam: A method for stochastic optimization. In *International Conference on Learning Representations (ICLR)*, San Diego, CA, USA.
- Köpüklü, O., Wei, X., and Rigoll, G. (2019). You only watch once: A unified cnn architecture for real-time spatiotemporal action localization. *arXiv preprint arXiv:1911.06644*.
- Li, Y., Chen, L., He, R., Wang, Z., Wu, G., and Wang, L. (2021). Multisports: A multi-person video dataset

- of spatio-temporally localized sports actions. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 13536–13545.
- Li, Y., Wang, Z., Wang, L., and Wu, G. (2020). Actions as moving points. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XVI 16*, pages 68–84. Springer.
- Lin, T.-Y., Dollár, P., Girshick, R., He, K., Hariharan, B., and Belongie, S. (2017). Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2117–2125.
- Liu, H., Li, C., Wu, Q., and Lee, Y. J. (2024). Visual instruction tuning. *Advances in neural information processing systems*, 36.
- Mandadapu, P. (2024). The evolution of football betting—a machine learning approach to match outcome forecasting and bookmaker odds estimation. *arXiv preprint arXiv:2403.16282*.
- Martens, F., Dick, U., and Brefeld, U. (2021). Space and control in soccer. *Frontiers in Sports and Active Living*, 3:676179.
- Martin, P.-E. (2020). *Fine-grained action detection and classification from videos with spatio-temporal convolutional neural networks: Application to Table Tennis*. PhD thesis, Université de Bordeaux.
- Redmon, J. (2016). You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*.
- Rudolph, Y. and Brefeld, U. (2022). Modeling conditional dependencies in multiagent trajectories. In *Proceedings of The 25th International Conference on Artificial Intelligence and Statistics*, pages 10518–10533. PMLR. ISSN: 2640-3498.
- Saha, S., Singh, G., Sapienza, M., Torr, P. H., and Cuzzolin, F. (2016). Deep learning for detecting multiple space-time action tubes in videos. *arXiv preprint arXiv:1608.01529*.
- Singh, G., Choutas, V., Saha, S., Yu, F., and Van Gool, L. (2023). Spatio-temporal action detection under large motion. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 6009–6018.
- Singh, G., Saha, S., Sapienza, M., Torr, P. H., and Cuzzolin, F. (2017). Online real-time multiple spatiotemporal action localisation and prediction. In *Proceedings of the IEEE international conference on computer vision*, pages 3637–3646.
- Soomro, K., Zamir, A. R., and Shah, M. (2012). UCF101: A dataset of 101 human actions classes from videos in the wild. *CoRR*, abs/1212.0402.
- Tomei, M., Baraldi, L., Calderara, S., Bronzin, S., and Cucchiara, R. (2021). Video action detection by learning graph-based spatio-temporal interactions. *Computer Vision and Image Understanding*, 206:103187.
- Vo-Ho, V.-K., Le, N., Kamazaki, K., Sugimoto, A., and Tran, M.-T. (2021). Agent-environment network for temporal action proposal generation. In *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2160–2164. IEEE.
- Wang, P., Bai, S., Tan, S., Wang, S., Fan, Z., Bai, J., Chen, K., Liu, X., Wang, J., Ge, W., et al. (2024). Qwen2-vl: Enhancing vision-language model’s perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*.
- Wang, Y., Sun, Y., Liu, Z., Sarma, S. E., Bronstein, M. M., and Solomon, J. M. (2019). Dynamic graph cnn for learning on point clouds. *ACM Transactions on Graphics (tog)*, 38(5):1–12.
- Weinzaepfel, P., Harchaoui, Z., and Schmid, C. (2015). Learning to track for spatio-temporal action localization. In *Proceedings of the IEEE international conference on computer vision*, pages 3164–3172.
- Zhang, Y., Tokmakov, P., Hebert, M., and Schmid, C. (2019). A structured model for action detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9975–9984.
- Zhao, J., Zhang, Y., Li, X., Chen, H., Shuai, B., Xu, M., Liu, C., Kundu, K., Xiong, Y., Modolo, D., et al. (2022). Tuber: Tubelet transformer for video action detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13598–13607.