DeepSoccer: Real-Time Object Detection and Tracking in Football Videos

Aditya Rathor[†], Vishesh Sachdeva[†], Aditya Mundhara[†], Siddhartha Singh[$]
[†]Department of Computer Science and Engineering, Indian Institute of Technology Jodhpur, India
[$]Department of Bioscience and Bioengineering, Indian Institute of Technology Jodhpur, India
Email: {b22ai044, b22ai050, b22ai057, b22bb041}@iitj.ac.in
Github: https://github.com/adityarathor007/DeepSoccer

***Abstract*—In this work, we present DeepSoccer, a deep learning-based framework for analyzing football videos using YOLO for real-time object detection and tracking. Our approach extends beyond detection by incorporating 2D projections, area matching, and spatial analysis to enhance player and ball movement understanding. These techniques enable improved scene interpretation, facilitating applications in game analytics and tactical assessment. Experimental results demonstrate the effectiveness of our approach in accurately tracking multiple entities and extracting meaningful insights from football matches.**

I. Introduction

Football is one of the most popular sports globally, with extensive applications in game analytics, tactical assessment, and performance evaluation. The ability to analyze player and ball movements in real-time provides valuable insights for coaches, analysts, and enthusiasts. Traditional methods for tracking player movements rely on expensive hardware, such as GPS trackers and motion capture systems. However, recent advancements in deep learning and computer vision have enabled automated tracking using only video footage.

In this work, we present DeepSoccer, a deep learning-based framework that leverages YOLO [1] (You Only Look Once) for real-time object detection and tracking in football videos. Our system not only detects and tracks key entities such as players, referees, and the ball but also extends its functionality to 2D projections and area matching. These additional features allow us to analyze spatial relationships between objects, providing deeper insights into gameplay dynamics.

By utilizing state-of-the-art deep learning models, our approach ensures high accuracy and efficiency, making it suitable for real-time applications. The proposed system can be used for various applications, including player performance evaluation, tactical planning, and automated video summarization. Our experimental results demonstrate the effectiveness of DeepSoccer in extracting meaningful information from football videos, highlighting its potential for both professional and recreational use.

II. Design

As discussed in [2], computer vision techniques can be applied to track football players effectively. Our system utilizes Ultralytics YOLOv11x and YOLOv11l for object detection in football videos, identifying four key classes: ball, referee, players, and goalkeeper. The models are trained on a diverse dataset to ensure robustness in varying camera angles, lighting conditions, and occlusions.

We first employed Ultralytics YOLOv11x to detect and classify players, referees, goalkeepers, and the ball in football videos. The model achieved strong detection performance as shown in Figure 1, with an overall mAP50–95 score of 0.614. Player detection performed the best, achieving 0.793 mAP50–95, followed by the goalkeeper (0.672) and referee (0.654). However, ball detection proved to be the most challenging, with a lower mAP50–95 of 0.336, likely due to its small size and rapid movement. Despite this, the model maintained high precision (0.965) for the ball, indicating accurate predictions when detected. These results highlight the need for further refinement in ball tracking.

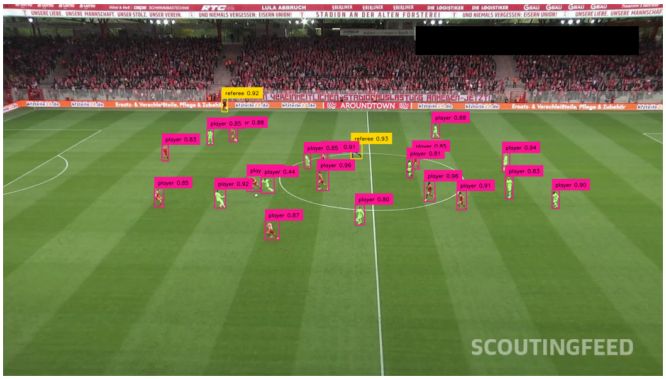

Fig. 1. Detection of Players in a Frame

To improve ball detection, we fine-tuned YOLO11l specifically for ball tracking by removing all other classes and retaining only the ground truth bounding boxes of the ball. This approach resulted in a notable improvement, increasing the mAP50–95 score from 0.336 to 0.416, while mAP50 improved from 0.707 to 0.813. Although precision slightly decreased to 0.918, recall significantly improved to 0.749, indicating better localization of the ball across frames. This enhancement suggests that training a specialized model for small, fast-moving objects like the ball can yield better de-

tection performance, making it more suitable for downstream tracking and analysis.

Next, we applied YOLOv11x specifically for ball tracking, removing all other classes to focus solely on the ball. This resulted in further improvements, with mAP50–95 increasing to 0.447 and mAP50 reaching 0.842, outperforming the YOLOv11l model. Precision slightly improved to 0.939, while recall decreased to 0.711, indicating that the model made more confident predictions but missed some detections. The higher parameter count and computational complexity of YOLOv11x contributed to better localization, making it a more suitable choice for tracking the ball accurately in football videos.

The improvement in ball tracking arises from enhanced feature extraction for small objects, reduced class imbalance, and better localization. By focusing solely on the ball, the model avoids down-weighting its bounding box coordinates in the loss function, leading to more precise gradient updates and improved detection performance.

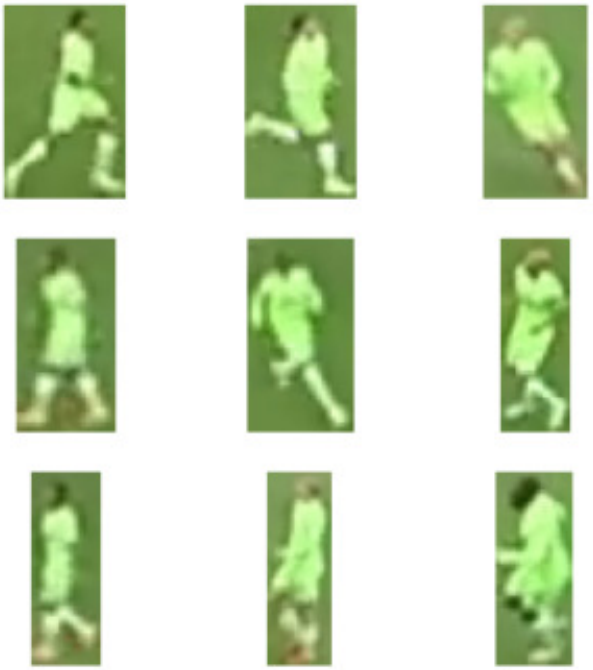

Fig. 2. Crops of Bounding Boxes of Players

We implemented a team clustering approach using player embeddings extracted from the SigLIP [3] model applied to cropped bounding boxes of detected players. These high-dimensional embeddings were reduced using UMAP [4] for effective representation in a lower-dimensional space, followed by K-Means clustering to distinguish the two teams, as shown in Figure 2. To further refine the clustering, we resolved goalkeeper team assignments based on spatial proximity. Assuming that goalkeepers are positioned closer to their team's defensive line, we computed the average positions of both team clusters and assigned goalkeepers to the nearest team. This method ensured accurate team classification and was implemented efficiently using PyTorch for seamless integration with our detection and tracking pipeline.

We employed ByteTrack [5] for object tracking in football videos, leveraging its robust association mechanism to handle fast-moving objects and frequent occlusions, as shown in Figure 3. The tracker was configured with a lost track buffer of 50, allowing objects to be temporarily lost and reacquired, which is crucial for handling rapid player movements and momentary occlusions. The minimum consecutive frames parameter was set to 1, enabling immediate track initialization to prevent delays in detection continuity. Additionally, a minimum matching threshold of 1.2 was used to maintain high tracking accuracy by filtering low-confidence associations.

Fig. 3. Tracking in football video

ByteTrack outperforms SORT and DeepSORT in football tracking due to its superior handling of low-confidence detections. Unlike SORT, which relies solely on high-confidence detections, ByteTrack incorporates both high- and low-confidence detections for robust association, reducing identity switches and improving long-term tracking. Compared to DeepSORT, which depends on appearance features for re-identification, ByteTrack is more suitable for football scenarios where players wear similar jerseys and undergo frequent occlusions, making appearance-based tracking less reliable.

To extract reference key points for pitch detection, as shown in Figure 4, we employed a fine-tuned YOLO pose estimation model, specifically trained to identify essential pitch landmarks, as shown in Figure 5. The model was applied to individual frames extracted from the input video using an iterator-based frame generator, enabling frame-wise processing. Given a selected frame (frame 532), the pose detection model was inferred with a confidence threshold of 0.3, generating multiple key points. To enhance reliability, a post-processing filter was applied, retaining only key points with a confidence score greater than 0.5, thereby reducing noise and ensuring precision. The filtered key points were then visualized using Supervision's VertexAnnotator, providing a clear representation of the detected pitch reference points. This method plays a crucial role in 2D-to-3D projection and pitch localization, forming the foundation for accurate spatial analysis and trajectory estimation in football tracking.

A homography transformation was employed to establish a spatial correspondence between the detected football pitch in the video frame and a standardized 2D pitch model. The Soc-

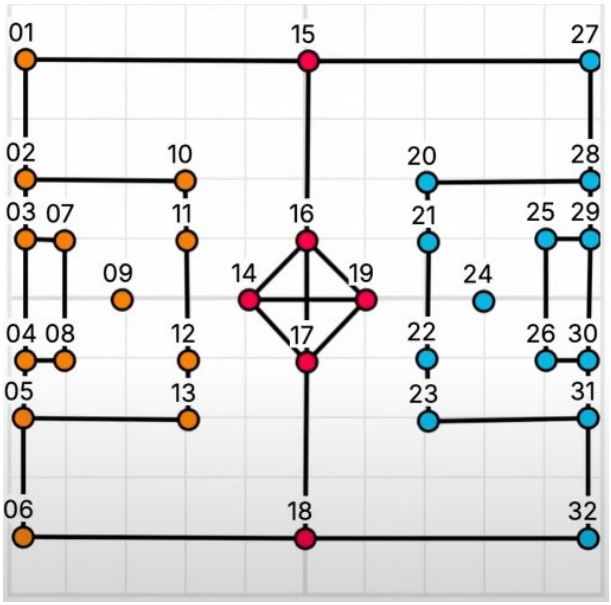

Fig. 4. Reference keypoints

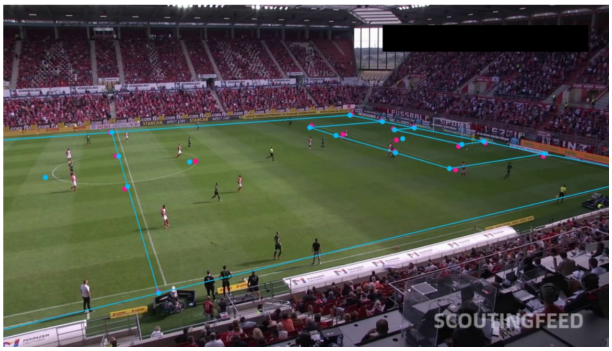

Fig. 5. Pitch Keypoint Detection

cerPitchConfiguration module defines 32 reference points on the standardized pitch, along with their connectivity structure, serving as the target plane for homography estimation. The source plane consists of key points detected within the video frame, filtered based on a confidence threshold of 0.5 to retain only reliable coordinates.

A ViewTransformer module was utilized to compute the homography matrix using OpenCV's findHomography function, which estimates a perspective transformation between the two sets of corresponding points. This transformation was then applied to project all points from the 2D pitch model onto the video frame, ensuring accurate pitch alignment. The transformed pitch layout serves as a foundational structure for downstream tasks such as player tracking, event detection, and spatial analysis.

To facilitate real-time 2D visualization, as shown in Figure 6, all detected entities—including players, referees, and the ball—were projected onto the standardized pitch representation. The Supervision Detections module was used to extract bounding boxes for these entities, which were then

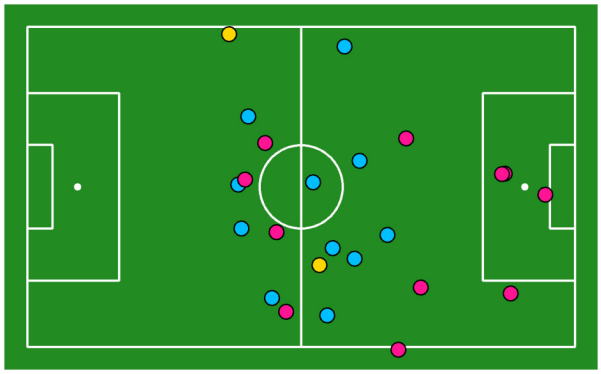

Fig. 6. 2D Pitch Projection

transformed using the computed homography matrix.

Players were classified into two teams using a trained TeamClassifier, which assigned team labels based on cropped player detections. Goalkeepers were further classified by resolving their team identities through spatial proximity analysis with their respective teammates.

Referees were identified separately and visually distinguished from players. The ball was assigned a unique class and projected onto the pitch using its detected coordinates.

For enhanced clarity, multiple annotation techniques were employed:

- Supervision's Vertex and Edge Annotators were used to overlay detected pitch key points (in pink) and the transformed pitch layout (in blue).
- Ellipse-based markers were used to represent projected player positions, with team-based color coding (blue and pink).
- A triangle annotation was applied to highlight the ball's position on the 2D pitch.

This approach provides an interpretable representation of player movements and ball trajectory, allowing for in-depth match analysis, tactical evaluations, and event detection in football tracking.

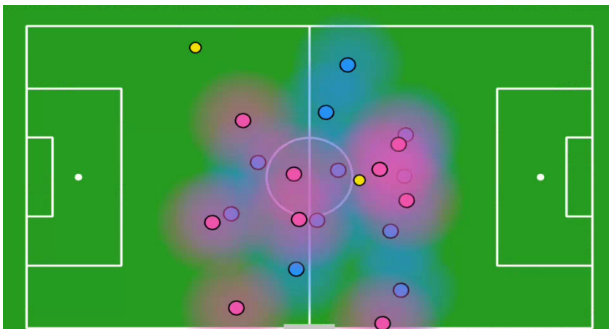

Fig. 7. Control Area of players

III. Conclusion and Limitations

In this work, we developed a framework for analyzing football gameplay using computer vision techniques. Our

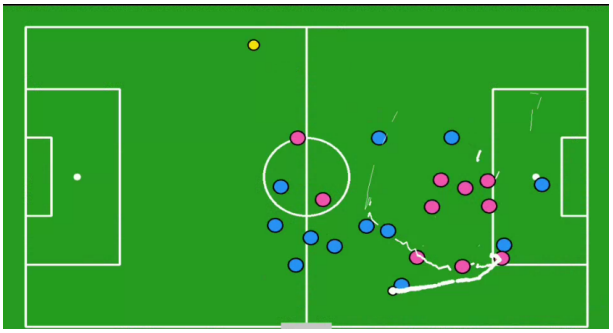

Fig. 8. Ball tracking

approach included object detection, object tracking, pitch landmark detection, and homography transformation, allowing us to project detected objects from the camera view onto a standardized 2D pitch representation. The dataset used in this study was obtained from Roboflow [6], [7].

Using this method, we visualized key aspects of the game, including:

- Player control areas, as shown in Figure 7, to understand spatial coverage and positioning.
- Ball trajectory tracking, as shown in Figure 8, to analyze movement patterns and shot accuracy.
- Ball possession analysis, to determine which team controlled the ball.
- Pass detection and accuracy, to evaluate the number of passes and their success rates.
- These visualizations provide valuable insights for match analysis, player performance evaluation, and tactical assessments.

Despite the effectiveness of our approach, certain limitations affect the accuracy and robustness of our analysis:

- Lofted Shot Projection Inaccuracy – During lofted shots, our 2D projection method represents the ball trajectory as a curved path on the pitch. Since this does not account for vertical displacement, inaccuracies may arise in ball possession tracking and related gameplay analyses.
- Lack of Ground-Truth Data for Object Tracking Evaluation – Our evaluation is limited by the absence of a benchmark dataset for object tracking in football videos. This makes it challenging to quantitatively assess the accuracy and robustness of our tracking techniques.
- Occlusion, Player Overlaps, and High-Speed Ball Motion – In crowded scenarios, especially near the goal area, players frequently overlap or occlude one another, leading to misidentification and tracking errors. Additionally, the ball moves at high speeds during fast plays, making it difficult to track efficiently, which can impact the accuracy of ball trajectory estimation and possession analysis.

IV. Contributions

- **Aditya Rathor** worked on team clustering, pitch keypoint detection and assigning goalkeepers to the team.
- **Vishesh Sachdeva** worked on ball-specific model fine-tuning and tracking, ball trajectory, ball possession, total passes.
- **Aditya Mundhara** worked on homography conversion and 2D projection.
- **Siddhartha Singh** worked on Object tracking and detection.

References

[1] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," 2016.

[2] P. Skalski, "Track football players with computer vision." Roboflow Blog, Dec. 9 2022.

[3] X. Zhai, B. Mustafa, A. Kolesnikov, and L. Beyer, "Sigmoid loss for language image pre-training," 2023.

[4] L. McInnes, J. Healy, and J. Melville, "Umap: Uniform manifold approximation and projection for dimension reduction," 2020.

[5] Y. Zhang, P. Sun, Y. Jiang, D. Yu, F. Weng, Z. Yuan, P. Luo, W. Liu, and X. Wang, "Bytetrack: Multi-object tracking by associating every detection box," 2022.

[6] Roboflow, "football-field-detection dataset." https://universe.roboflow.com/roboflow-jvuqo/football-field-detection-f07vi, aug 2024. visited on 2025-03-31.

[7] Roboflow, "football-players-detection dataset." https://universe.roboflow.com/roboflow-jvuqo/football-players-detection-3zvbc, mar 2025. visited on 2025-03-31.