SOFA - Step-On-Foot Analyzer

Federico Gerardi
MSc Computer Science
Sapienza University of Rome
gerardi.1982783@studenti.uniroma1.it

Murad Hüseynov
MSc Computer Science
Sapienza University of Rome
huseynov.2181584@studenti.uniroma1.it

***Abstract*—A step-on-foot foul in football occurs when a player steps on an opponent's foot, restricting their movement and causing discomfort. This type of foul is considered objective, as it does not rely on subjective interpretation but is clearly penalized when it occurs. However, referees face challenges in detecting such fouls in real-time due to the fast pace of the game. While the use of VAR (Video Assistant Referee) can assist in reviewing incidents, it is time-consuming and disrupts the match flow. Additionally, inconsistencies in foul interpretations across different referees and competitions create further uncertainty in decision-making.**

In this work, we propose an automated approach for step-on-foot foul detection using a deep learning-based model ensemble. Our system integrates YOLO for keypoint detection, Depth-Anything for player filtering, and SAM2 for segmentation, enabling precise tracking of foot interactions. A weighted overlap calculation method prioritizes foot contact in the lower regions of the detected masks, effectively reducing false positives from leg interactions. Furthermore, peak detection and temporal filtering techniques ensure robust foul detection across video frames. The system processes replay footage of potential fouls and outputs an overlap percentage, allowing for the establishment of an optimal threshold to maximize the F1-score in foul detection. This approach offers a consistent, efficient, and automated solution for improving foul detection accuracy in football, reducing reliance on subjective human interpretation and minimizing game interruptions due to VAR.

I. Introduction

Football referees play a crucial role in ensuring fair play by enforcing the rules of the game, yet some infractions remain difficult to detect in real-time. One such instance is the **step-on-foot foul**, which occurs when a player steps on an opponent's foot, impeding their movement or causing discomfort. This type of foul is particularly challenging to identify due to the fast-paced nature of the sport and the positioning of players on the field. Despite the assistance of **Video Assistant Referee (VAR) technology**, inconsistencies in decision-making persist, as fouls are still subject to human interpretation and review delays that disrupt the flow of the match.

An objective and automated approach to detecting step-on-foot fouls could greatly enhance decision-making consistency in football. Existing solutions rely on referee observations, slow-motion replays, and VAR interventions, all of which introduce delays and subjectivity into the review process. Additionally, different referees may interpret similar situations differently, leading to inconsistency across games and competitions. A reliable foul detection system that objectively identifies these fouls based on quantifiable evidence could mitigate these issues.

In this paper, we present an **AI-driven model ensemble** designed to automatically detect step-on-foot fouls from match footage. The system combines **YOLO for keypoint detection**, **Depth-Anything for player selection**, and **SAM2 for segmentation and tracking**. To reduce false positives caused by leg overlap, we introduce a **weighted overlap calculation**, which prioritizes lower mask regions, ensuring that only **foot-to-foot** contacts are emphasized. Our approach provides **an automated**, **efficient**, and **objective foul detection mechanism**, reducing reliance on subjective human interpretation and minimizing disruptions caused by manual VAR reviews.

By developing this system, we aim to contribute to the field of **sports analytics and AI-assisted refereeing**, offering a tool that enhances the accuracy and consistency of foul detection in football. Our methodology ensures that only meaningful contact between players' feet is analyzed, leading to **improved decision-making**, **fairer gameplay**, and **reduced stoppages** in professional matches.

II. Related Work

A. Existing Work on Automated Foul Detection in Football

Automated foul detection in football has historically relied on rule-based methods and manual video reviews. The Video Assistant Referee (VAR) system [8] exemplifies current practices, yet it remains hampered by inherent subjectivity and review delays that can disrupt match flow. Early computer vision approaches attempted to codify heuristics for detecting fouls, but these methods were unable to cope with the dynamic and complex nature of live gameplay. Recent advances in deep learning, however, have opened new avenues for real-time, objective analysis. In particular, leveraging end-to-end

trainable models enables the extraction of discriminative features directly from video data, paving the way for automated systems that can complement or even enhance traditional officiating.

B. YOLO and Pose Estimation in Sports Analytics

Real-time object detection is critical in sports analytics, where rapid player interactions must be captured accurately. Our work utilizes YOLOv11 [1], the latest advancement in the YOLO series, which offers significant improvements in speed and accuracy over its predecessors. YOLOv11 excels at detecting players and localizing keypoints, such as the feet, which are crucial for identifying step-on-foot fouls. In addition, complementary pose estimation methods, such as those described by Cao et al. [2], provide detailed joint localization, ensuring that subtle interactions between players are accurately monitored. This combination of fast detection and fine-grained keypoint analysis is essential for differentiating between incidental contacts and actual fouls.

C. Depth Estimation for Player Filtering

Accurate identification of the players involved in a foul is challenging, particularly in crowded scenes. To address this, our approach incorporates Depth-Anything V2 [3]—a state-of-the-art model for monocular depth estimation. Depth-Anything V2 generates reliable depth maps that add crucial spatial context, allowing the system to filter out players who are not in close proximity to the event. By leveraging depth cues, our framework effectively reduces false positives, ensuring that only the players in the immediate area of a potential foul are analyzed further. This additional layer of spatial filtering is critical in complex environments where overlapping players can otherwise confuse the detection pipeline.

D. Segmentation for Object Tracking in Sports

Robust segmentation is key to isolating the regions of interest, particularly when distinguishing foot-to-foot contacts from other body interactions. While traditional segmentation networks like Mask R-CNN [4] and DeepLab [5] have been widely used, they are less adept at handling the temporal dynamics and visual variability present in sports videos. To overcome these challenges, we employ Segment Anything 2 (SAM2) [6]. SAM2 is specifically designed to extend the capabilities of its predecessor to video data, offering improved generalization across diverse scenes. Its ability to accurately segment the lower regions of players—where critical foul interactions occur—makes it an ideal choice for our system. This precision in segmentation is pivotal for our weighted overlap calculation, which prioritizes genuine foot contacts over incidental leg interactions.

E. Other AI Applications in Sports Officiating

The trend toward AI-driven officiating extends well beyond foul detection. Deep learning models have been successfully applied to a variety of tasks, such as event recognition and performance analysis, in sports analytics. Datasets like SoccerNet [7] have facilitated the training of robust models capable of detecting and classifying key events (e.g., goals, offsides, fouls) in football matches. These advancements illustrate a broader shift in sports technology towards systems that reduce human error, enhance consistency, and support real-time decision-making. Our work builds on this foundation by targeting a specific, yet challenging, aspect of foul detection with a finely tuned ensemble of deep learning models.

III. Proposed Method

In this section, we describe our automated pipeline for detecting step-on-foot fouls. Our approach consists of four main stages: *(A)* Player Keypoint Detection using YOLO, *(B)* Depth Filtering with Depth-Anything, *(C)* Foot Segmentation & Tracking via SAM2, and *(D)* Weighted Overlap Calculation for foul assessment.

A. Player Keypoint Detection

The primary goal of this stage is to detect players and locate their foot keypoints in a given video frame. We focus on each player's left and right ankle/foot regions, as these are critical for determining foot-to-foot contact.

1) **YOLO Pose Model.** We utilize a YOLO v11 Pose model to perform real-time detection and pose estimation. Instead of merely bounding players as whole objects, the pose variant of YOLO provides keypoints for each detected player, including ankles and feet

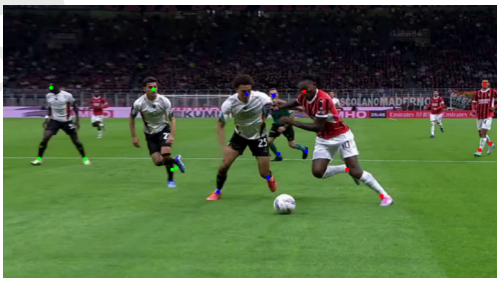

Fig. 1: *Keypoint detection using YOLO*

2) **Greedy Keypoint Extraction.** We run YOLO on a designated frame of interest (e.g., a replay segment suspected of containing a foul). The model outputs multiple sets of keypoints—one for each detected player. In practice, this yields the coordinates $(x_{leftFoot}, y_{leftFoot})$ and $(x_{rightFoot}, y_{rightFoot})$ for each player, along with other body joints that can be used for reference.

```
for person in results[0].keypoints.xy:
    left_foot = person[15].cpu()
    right_foot = person[16].cpu()
    ...
```

Listing 1: *YOLO's pose model indexes body joints in a fixed order. Index 15 corresponds to the left ankle and index 16 corresponds to the right ankle.*

B. Depth Filtering

Once keypoints are detected, we must identify which two players are most likely involved in the foul. In many match scenarios, multiple players appear in the camera view, but only two are relevant (the one committing the foul and the one receiving it).

1) **Depth-Anything.** To isolate the relevant players, we adopt a state-of-the-art monocular depth estimation model, Depth-Anything V2. For each detected player, we query the depth value at a reliable keypoint location (e.g., the nose) to approximate how close the player is to the camera.
2) **Nearest Players Selection.** We sort all detected players by their estimated depth and retain only the top k closest individuals (typically $k = 2$). These two players are assumed to be the ones in direct contact, thus reducing noise from other on-screen players.

```
pipe = pipeline(task="depth-estimation", model
    ="depth-anything/Depth-Anything-V2-Large-
    hf")

depth = np.array(pipe(Image.fromarray(frame))[
    "depth"])

for person in people:
    x_nose = int(person["nose"][0])
    y_nose = int(person["nose"][1])
    person["depth"] = depth[y_nose, x_nose]

people = sorted(people, key=lambda x: x["depth
    "], reverse=True)
```

Listing 2: *Depth-Anything filtering computes a depth map at each nose keypoint, then retains the two closest players.*

C. Foot Segmentation & Tracking

Accurately modeling the foot region is crucial for distinguishing foot-to-foot contact from incidental overlap of legs or other body parts. To achieve fine-grained segmentation of each foot over time, we employ SAM2 (Segment Anything 2) in video mode.

1) **Frame Extraction.** First, we decompose the input replay into individual frames. For each video, frames are stored in a numbered sequence (e.g., *00000.jpg*, *00001.jpg*, ...).
2) **Initial Foot Bounding Boxes.** We place a small bounding box around each foot keypoint. These bounding boxes serve as prompts to SAM2, indicating the approximate location of the foot.
3) **SAM2 Initialization & Propagation:**
   - **Initialization.** On a designated "annotation" frame (e.g., the same frame used for keypoint detection), SAM2 generates an initial foot mask for each bounding box prompt.
   - **Propagation.** SAM2 then propagates these foot masks forward and backward across all frames in the sequence, producing a set of foot segmentation masks for each frame

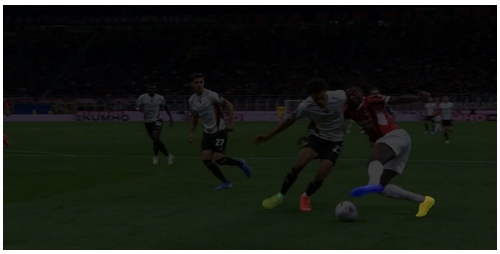

Fig. 2: *Sample foot mask created with SAM2*

4) **Optional Refinement**. If needed, additional positive or negative clicks are supplied to refine the segmentation on the annotation frame. The updated masks are once again propagated throughout the video, ensuring consistent and accurate foot regions.

D. Weighted Overlap Calculation for Foul Assessment

Having per-frame segmentation masks of each player's feet, the final step is to quantify foot-to-foot contact.

1) **Rationale for Weighted Overlap.** Naively computing the pixel intersection between two feet can lead to false positives when legs overlap at the thigh or shin level. We address this by assigning higher weights to lower rows in the segmentation mask—i.e., the part of the foot that is physically in contact with the ground.
2) **Weight Matrix.** We construct a weight matrix W where the weight for row i increases from top to bottom. This captures the intuition that contact near the bottom of the foot is a stronger indicator of a genuine step-on-foot foul.

$$W_{i,j} = \frac{i}{H}, \quad \forall\, 0 \le i < H,\, 0 \le j < W, \tag{1}$$

3) **Overlap Computation.** For each frame f:
   a) Retrieve the binary foot masks M_1 and M_2 for the two players.
   b) Compute the intersection mask $\mathcal{I} = (M_1 \wedge M_2)$.
   c) Calculate weighted overlap as $\sum_{(i,j)\in\mathcal{I}} W_{i,j}$.
4) **Threshold & Temporal Filtering.** We define an overlap threshold τ (e.g., $\tau = 30$ and require that τ be exceeded for a specified number of consecutive frames (e.g., 2 frames) to declare a potential foul. This temporal filtering helps avoid spurious one-frame overlaps caused by motion blur or segmentation errors.

```
h, w = sample_mask.shape[:2]
weights = (np.arange(1, h + 1) / h).reshape(h,
    1)
weight_matrix = np.tile(weights, (1, w))

intersection = mask1 & mask2
weighted_overlap = np.sum(weight_matrix *
    intersection)
frame_overlap += weighted_overlap
```

Listing 3: *Each row i is given a weight proportional to $\frac{i}{H}$, emphasizing contacts at the bottom of the foot. The final overlap score (`weighted_overlap`) accumulates in `frame_overlap`, driving the foul decision.*

E. Foul Detection Output

Finally, the system outputs:

1) **Overlap Scores per Frame.** For each video frame, the weighted overlap is recorded, providing a trace of foot contact over time.
2) **Binary Foul Flag.** If the overlap exceeds the threshold τ for the required number of consecutive frames, the system flags a "potential step-on-foot foul."
3) **Segmented Replay.** (Optional) The user may export a replay video overlaid with color-coded foot segments. This replay can be used for post-game analysis or as supportive evidence for referee decisions.

IV. Dataset

Our dataset is constructed using DAZN Serie A Highlights available on YouTube. We conducted a thorough search to identify matches that potentially included incidents of stepping on a player's foot. After identifying these matches, we downloaded the relevant highlight videos. We then carefully edited these highlights to isolate the specific moments of interest. Finally, we prepared these edited clips for testing with our algorithm to analyze and detect stepping incidents.

Unfortunately, our dataset is limited in size due to several constraints. We do not have access to all the replays of potential step-on-foot fouls, which restricts the number of incidents we can analyze. Additionally, we lack complete replays of every game, where such incidents typically occur about 5 to 6 times per match on average. This limitation significantly impacts the comprehensiveness of our dataset and our ability to thoroughly test and validate our algorithm for detecting step-on-foot fouls.

As part of our future work, we aim to establish a collaboration with a Football Federation. By partnering with such an organization, we will gain access to a more extensive and diverse set of match replays and highlights. This collaboration will significantly enhance our dataset, allowing us to include a broader range of step-on-foot incidents. As a result, our model will benefit from improved training data, leading to better accuracy and reliability in detecting these specific fouls. This strategic partnership is crucial for advancing our research and ensuring that our algorithm performs optimally in real-world scenarios.

V. Benchmark

Our algorithm demonstrates excellent performance in minimizing false positives; however, we are experiencing a high number of false negatives. While our F1-Score is currently lower than that of VAR, it still outperforms standard referees. Additionally, our model operates four times faster than VAR when making decisions, making it a valuable tool to assist VAR in reaching decisions more quickly.

Precision	Recall	F1-Score	Avg. Time
0.83	0.45	0.58	1 min

TABLE I: SOFA Metrics

Precision	Recall	F1-Score	Avg. Time
0.875	0.77	0.81	4 mins

TABLE II: VAR Metrics

VI. Conclusions

We have developed an innovative tool aimed at assisting referees in making faster and more objective decisions during matches. However, we acknowledge that the tool is not yet flawless; it still requires human oversight to confirm whether a foul has actually occurred. Our ultimate goal is to refine and enhance this algorithm to the extent that it can autonomously make accurate decisions, minimizing the need for referee intervention. By achieving this, we hope to improve the overall efficiency and fairness of officiating in sports, allowing referees to focus on other critical aspects of the game while relying on our tool for support.

VII. Future Work

We are committed to addressing the existing limitations in our system, particularly the necessity for the camera to maintain a clear focus on the players involved in a foul. This requirement can be challenging, especially in noisier video environments where multiple actions are occurring simultaneously. To improve our system's performance, we aim to enhance its ability to accurately detect and analyze fouls even in these more complex scenarios.

Furthermore, we plan to introduce the concept of a step on foot foul that involves more than two players. This will allow us to capture a broader range of interactions on the field, providing a more comprehensive understanding of the dynamics at play during a foul situation. By implementing this feature, we hope to create a more robust and versatile system that can effectively handle various game situations, ultimately improving the accuracy and reliability of our foul detection capabilities.

References

[1] You Only Look Once v11 (YOLOv11) https://arxiv.org/pdf/2410.17725.
[2] Cao, Z., Simon, T., Wei, S. E., & Sheikh, Y. (2017). Real-time Multi-Person 2D Pose Estimation using Part Affinity Fields. https://arxiv.org/abs/1611.08050
[3] Depth-Anything 2 https://arxiv.org/pdf/2406.09414
[4] He, K., Gkioxari, G., Dollár, P., & Girshick, R. (2017). Mask R-CNN. https://arxiv.org/abs/1703.06870
[5] Chen, L.-C., Papandreou, G., Kokkinos, I., Murphy, K., & Yuille, A. L. (2017). DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs. https://arxiv.org/abs/1606.00915
[6] Segment Anything 2 (SAM2) https://arxiv.org/abs/2408.00714
[7] Giancola, S., Amine, M., & Van Gool, L. (2018). SoccerNet: A Scalable Dataset for Action Spotting in Soccer Videos. https://arxiv.org/abs/1804.04527

[8] FIFA. (2018). Video Assistant Referee (VAR) System Operational Guidelines. https://inside.fifa.com/innovation/standards/video-assistant-referee