CSCI5561- S25- Final Report
Reconstructing the Game: A Low-Cost Multiview Soccer Play Analysis for Fair Officiating

James Sargsyan, Beñat Froemming-Aldanondo, Trevor Nguyen
University of Minnesota

Abstract

This project focuses on building a low-cost, multiview 3D reconstruction system to help analyze soccer plays and make officiating technology more accessible. Using limited multiview footage, we calibrate cameras with a template of a regulatory soccer pitch layout, detect player joint keypoints, and estimate their 3D positions in world coordinates. The goal is to assist referees in decisions like offsides, offering a more practical and affordable alternative to VAR. We test the system on real game footage and demonstrate that it is a feasible solution for lower-league or amateur matches where resources are limited.

1 Introduction and Motivation

Refereeing professional soccer games has been revolutionized with the advent of the Video Assistant Referee (VAR) system, which provides precise and reliable officiating through high-speed cameras and advanced tracking algorithms [FIFA, 2025]. In high-intensity and pivotal soccer plays, it is now a simple matter to review and resolve any disputes that may occur and continue the game. However, millions of amateur soccer enjoyers worldwide are still subject to biased human judgment and heated disputes since they do not have easy access to this technology.

One of the fundamental rules that causes many doubts in soccer is offsides due to its complexity. Offside occurs when an attacking player is closer to the opponent's goal line than both the ball and the second-last defender (to account for the goalie) at the moment the ball is played to them, unless they are in their own half. The current system employed in professional leagues for offsides reviewing is the Semi-Automated Offside Technology (SAOT) [FIFA, 2023]. It was first used in the 2022 FIFA World Cup in Qatar, where it was tested on a global stage and later implemented in the 2022-2023 UEFA Champions League and domestic leagues, including La Liga, Serie A, and Premier League. It uses 12 dedicated tracking cameras mounted underneath the roof of the stadium to track the ball and up to 29 data points of each individual player, 50 times per second, calculating their exact position on the pitch. The 29 collected data points include all limbs and extremities that are relevant for making offside calls. The referees can then generate a virtual offside plane as seen in Figure 1, improving decision speed and accuracy. However, the technology remains costly—reaching up to $5 million per stadium—and demands substantial computational resources, including numerous advanced cameras, high-performance AI systems, and real-time data processing capabilities, which make implementation and maintenance expensive.

Figure 1: Semi-Automated Offside Technology.

To address this challenge, our project proposes a low-cost, lightweight solution deployable with readily available equipment, such as phone cameras and tripods. The pipeline we developed reconstructs the game at moments of interest, and aids refereeing by providing reasonably accurate three-dimensional insights to support human officials. Using limited multi-view footage, we calibrate the cameras based on a template of a regulation soccer pitch, then detect player joint keypoints and estimate their positions in world coordinates.

2 Related Work

A significant amount of work on offside detection has been conducted, driven by soccer's global popularity and the substantial funding it attracts. Previous attempts at creating more affordable systems have largely relied on single-camera setups. To the best of our knowledge, this paper is the first to address multi-view reconstruction with limited resources in mind. The foundation of our project is inspired by several research papers within the same technological domain.

In the paper *Real Time Offside Detection using a Single Camera in Soccer* by Shounek Desai [Desai, 2025], filtering and edge detection techniques are used to estimate the vanishing point, the point at which parallel lines in 3D space, when viewed in perspective, appear to converge. A joint key point estimation algorithm is then used to identify an individual player's poses. Every keypoint can be connected with the vanishing point, and an offside line is simply the parallel line belonging to the last defender. If it is crossed by an opposing player's parallel line, it is an offside. The vanishing point gives some three-dimensional information in the 2D image.

In another study, Offside Detection for Better Decision-Making and Gameplay in Football [Madake et al., 2023], the authors also aim to detect offside events, but their approach emphasizes logical decision-making over precise spatial positioning. They begin by removing background noise and extracting features based on jersey color to differentiate between teams, utilizing Histograms of Oriented Gradients (HOG) for player detection. Once players are categorized by team, the algorithm estimates each player's x-coordinate in image space, appends these to lists, sorts them, and compares the positions of the leftmost and rightmost players to assess offside scenarios. This method achieved up to 98.5% accuracy in classifying players, but faced challenges with misclassifying referees and goalkeepers due to jersey color differences. Additionally, relying on image coordinates rather than true world coordinates limits the spatial accuracy of offside judgments.

Some other relevant research papers that we got inspiration from are the following: *Vision Based Dynamic Offside Line Marker for Soccer Games* [Muthuraman et al., 2018], *Offside Detection System using an Infrared Camera Tracking System* [Unknown, 2025], *An Investigation Into the Feasibility of Real-Time Soccer Offside Detection From a Multiple Camera System* [D'Orazio et al., 2009], *VARS: Video Assistant Referee System for Automated Soccer Decision Making from Multiple Views* [Held et al., 2023].

3 Proposed Approach

The first step of our pipeline involves obtaining high-quality multi-view soccer footage. Due to copyright restrictions, publicly available recordings of professional matches are limited. For this project, we use the SoccerNet dataset [SoccerNet, 2023], the largest public collection of soccer video data. SoccerNet provides a wide range of images and annotations for various tasks; we specifically use the action replay subset, which includes annotations for action spotting along with synchronized replay views of the same event. Each moment typically includes between two and four camera angles—fewer than the twelve-camera setups used in professional systems—making it a good fit for simulating a low-resource video assistant referee system. We selected set of images from a single side of the pitch since the pitch is symmetrical and the model can't distinguish the side since it only looks for pitch markers.

After obtaining the images, our next step is camera calibration. This involved estimating the intrinsic and extrinsic parameters for each camera view. For this, we use a method inspired by the SoccerNet Camera Calibration Challenge [Magera et al., 2024]. This approach leverages pitch markings, static elements common to all soccer fields, and aligns them with a known regulatory field template illustrated in Figure 2. The template represents the soccer pitch in world coordinates with origins placed at the center of the field. It assumes a flat plane on the xy axis with z = 0 the z axis facing downward, as seen in Figure 3

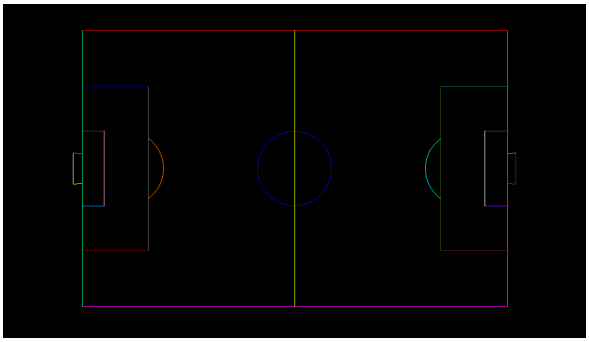

Figure 2: Pitch template viewed from above.

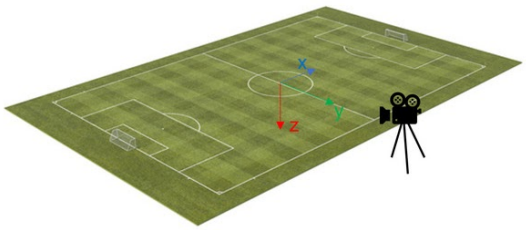

Figure 3: World coordinate system.

The next step in our pipeline is marker detection. To achieve this, we used the baseline segmentation model provided by the challenge, which classifies each pixel as one of 26 field markers or as background, as illustrated in Figure 4. However, converting the raw segmentation mask into usable line data required additional processing.

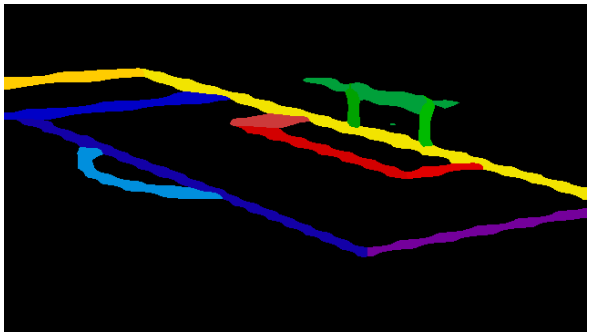

Figure 4: Raw mask before post-processing.

The post-processing involves several steps. First, we remove segments corresponding to goalposts and circular markers, as they are too noisy. Then, for each remaining label, we extract the largest contour to reduce noise and focus on the main structure. A line is fit to each contour and clipped according to the minimum and maximum pixel ranges. To improve continuity, we average and merged endpoints that are close to one another using a threshold.

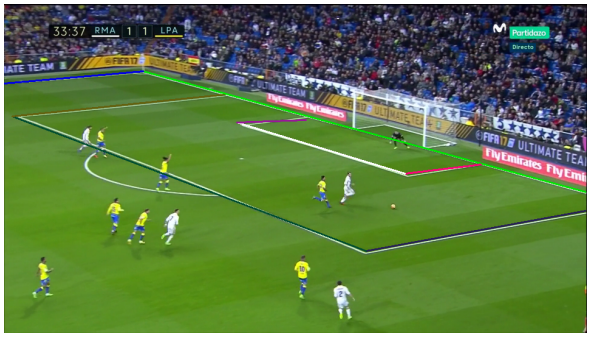

Figure 5: Post-processed mask.

Finally, we establish correspondences between the detected lines and the pitch template using the marker labels. With the line correspondences established, the next step in the pipeline is to compute a homography matrix H that maps the regulatory pitch lines, defined in real-world coordinates, to their corresponding 2D lines in the image. This process is similar to point-based homography estimation, where point correspondences satisfy:

$$\mathbf{x}' = H\mathbf{x}$$

For lines, however, the transformation is given by:

$$\mathbf{l}' \sim H^{-\top}\mathbf{l}$$

where $\mathbf{l}$ is a line in world (pitch) coordinates and $\mathbf{l}'$ is the corresponding line in the image, both represented in homogeneous form.

Each line correspondence provides two independent linear constraints on the homography H, contributing two rows to the constraint matrix $A \in \mathbb{R}^{2N\times 9}$ for N correspondences. Given a line correspondence $\mathbf{l} \leftrightarrow \mathbf{l}'$, with $\mathbf{l} = [l_1, l_2, l_3]^\top$ and $\mathbf{l}' = [l'_1, l'_2, l'_3]^\top$, the constraints can be written as:

$$A = \begin{bmatrix} \mathbf{0}_{1\times 3} & -l_3\mathbf{l}'^\top & l_2\mathbf{l}'^\top \\ l_3\mathbf{l}'^\top & \mathbf{0}_{1\times 3} & -l_1\mathbf{l}'^\top \end{bmatrix}$$

To avoid explicitly computing H^{-1}, we linearize the constraint using the Kronecker product and construct a system of equations of the form:

$$A\mathbf{h} = 0$$

where $\mathbf{h}$ is a 9-dimensional vector formed by stacking the entries of H. We obtain the homography estimated as the solution to this homogeneous system using Singular Value Decomposition (SVD), by selecting the right singular vector corresponding to the smallest singular value.

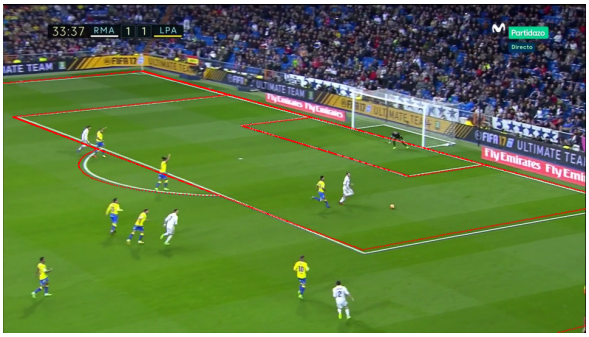

Figure 6: Template projected onto the image using H.

The homography between a planar scene and its image projection can be used to estimate both the camera's intrinsic and extrinsic parameters. This process follows Algorithm 8.2 from *Multiple View Geometry in Computer Vision* [Hartley and Zisserman, 2003]. We begin by estimating the intrinsic calibration matrix K, which encodes the focal lengths and the principal point of the camera.

Given a homography matrix $H \in \mathbb{R}^{3\times3}$, we reshape H into a vector and use it to construct a constraint matrix A based on the orthogonality and normalization conditions of the rotation components in H. Specifically, we enforce:

$$\mathbf{r}_1^\top B\mathbf{r}_2 = 0 \quad \text{and} \quad \mathbf{r}_1^\top B\mathbf{r}_1 = \mathbf{r}_2^\top B\mathbf{r}_2$$

where $B = K^{-\top}K^{-1}$ is a symmetric matrix representing the image of the absolute conic.

These conditions yield a homogeneous linear system $A\mathbf{w} = 0$, which is solved using Singular Value Decomposition (SVD). The resulting solution vector $\mathbf{w}$ defines the entries of matrix B, and we reconstruct K by applying a Cholesky decomposition:

$$B = K^{-\top}K^{-1} \quad \Rightarrow \quad K = \big(\text{Cholesky}(B^{-1})\big)^\top$$

Finally, the matrix K is normalized so that $K_{33} = 1$. The estimated parameters include the focal lengths f_x, f_y and the principal point (c_x, c_y), resulting in the intrinsic calibration matrix:

$$K = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix}$$

Once the intrinsic calibration matrix K has been estimated from a homography, we can recover the camera's extrinsic parameters, in specific its rotation R and position $\mathbf{t}$. This is based on Example 8.1 from the same book. Given a homography H and the calibration matrix K, we compute:

$$H' = K^{-1}H$$

This matrix H' approximates the first two columns of the camera's rotation matrix and a scaled translation vector. We extract the rotation columns:

$$\mathbf{r}_1 = \frac{\mathbf{h}'_1}{\|\mathbf{h}'_1\|}, \quad \mathbf{r}_2 = \frac{\mathbf{h}'_2}{\|\mathbf{h}'_2\|}$$

and compute the third column as their cross product:

$$\mathbf{r}_3 = \mathbf{r}_1 \times \mathbf{r}_2$$

These form a matrix $R = [\mathbf{r}_1\ \mathbf{r}_2\ \mathbf{r}_3]$, which we orthonormalize using SVD to ensure it is a valid rotation matrix. The translation vector $\mathbf{t}$ is obtained from the third column of H', scaled appropriately. Finally, the camera center $\mathbf{C}$ in world coordinates is given by:

$$\mathbf{C} = -R^\top \mathbf{t}$$

Once the intrinsic calibration matrix K and the extrinsic parameters (rotation matrix R and translation vector $\mathbf{t}$) are estimated, the final step is to construct the camera's projection matrix M. The projection matrix combines the intrinsic and extrinsic parameters to transform 3D world coordinates $\mathbf{X}$ into 2D image coordinates $\mathbf{x}$. The extrinsic parameters R and $\mathbf{t}$ are combined into a matrix $[R \mid \mathbf{t}]$, which represents the camera's orientation and position in the world. The projection matrix is then given by:

$$M = K[R \mid \mathbf{t}]$$

where M is a 3×4 matrix that maps 3D world points $\mathbf{X} = [X, Y, Z]^T$ to image coordinates $\mathbf{x} = [x, y]^T$ through the equation:

$$\mathbf{x} = P \begin{bmatrix} \mathbf{X} \\ 1 \end{bmatrix}$$

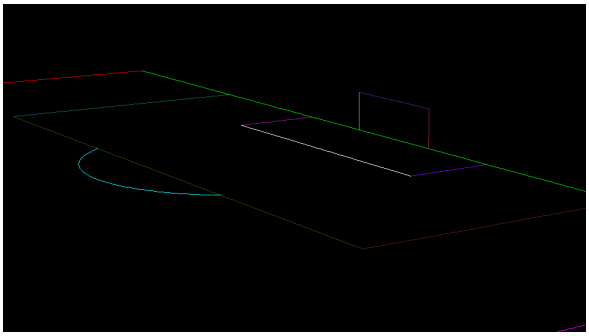

Figure 7: Camera view with estimated parameters.

With the cameras calibrated, the pipeline shifts focus to detecting players on the field. We employ a pretrained pose estimation model, Mask R-CNN [Abdulla, 2017], which detects 17 joint keypoints per person in image coordinates. These keypoints are then connected to form a skeleton representation of each player. To reduce noise from non-player detections, such as spectators, we manually filter the results and match keypoints to known players, assuming continuous tracking throughout the match. Limitations and ideas for automating this step are discussed in Section 4.

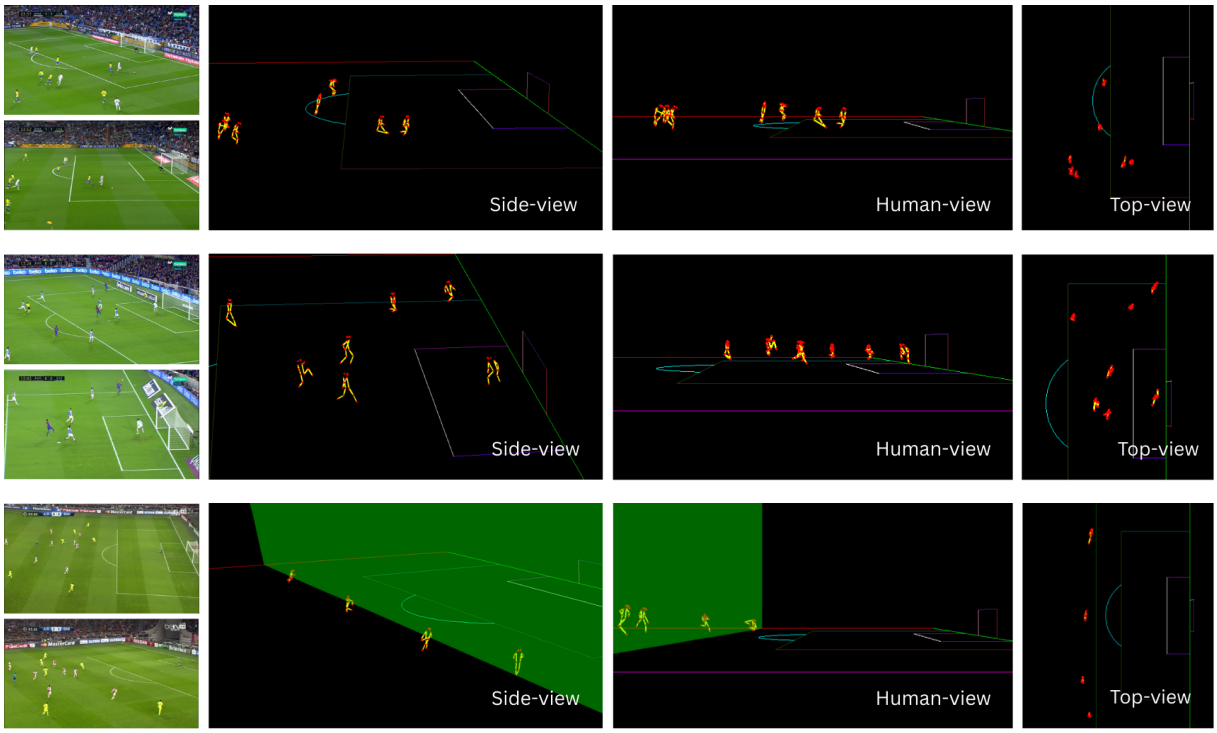

Figure 8: Sample raw images with corresponding reconstructions.

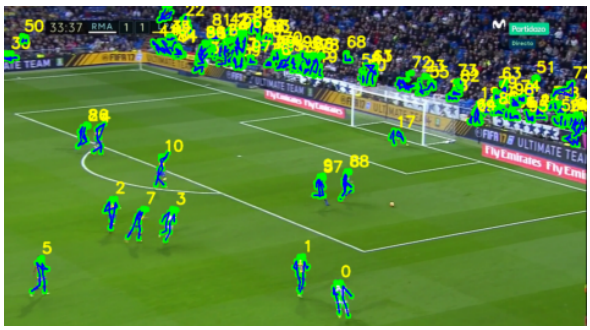

Figure 9: Pose estimations and skeletons.

Once we obtain joint correspondences and projection matrices for each camera, we use OpenCV's triangulation function to recover 3D keypoints in world coordinates. Given projection matrices P_1, P_2 and corresponding 2D points $\mathbf{x}_1, \mathbf{x}_2$, triangulation solves for the 3D point $\mathbf{X} \in \mathbb{P}^3$ that satisfies:

$$\mathbf{x}_1 \sim P_1 \mathbf{X}$$
$$\mathbf{x}_2 \sim P_2 \mathbf{X}$$

These points are visualized in a 3D pitch environment with skeleton overlays as seen in Figure 8. By combining data from multiple camera views, we reconstruct the scene from various angles. This reconstruction also enables analysis of tactical elements, such as drawing an offside plane to assess offside decisions.

4 Limitations and Future work

One of the primary limitations we encountered is the scarcity of usable multiview images in the SoccerNet dataset, which typically contains only 2–4 public views per match. Many of these images are either too low in quality or overly zoomed in, lacking visible pitch markers. Since our method relies on these markers to compute projection matrices, such images become unusable. This constraint significantly impacts the quality of the 3D reconstruction. Throughout the project, we realized that publicly accessible broadcast footage is limited due to copyright restrictions, making large-scale data collection challenging.

Our initial attempt used the SIFT algorithm [Lindeberg, 2012] to detect and match keypoints across views. From these matches, we aimed to estimate the fundamental or essential matrix, decompose it into relative pose (rotation and translation), and compute the projection matrices to triangulate 3D points. However, due to the complexity of the scenes and significant variation between viewpoints, standard methods like SIFT, ORB, and even deep learning-based feature matchers struggled to find reliable correspondences.

Regarding the pipeline, the pitch segmentation model we used produced noisy masks. Future improvements could involve fine-tuning this model

with additional data or training a new one from scratch. Enhancing the mask post-processing pipeline, especially by accounting for curved lines, would further improve accuracy. In terms of pose estimation, players often appear small in the frame, sometimes overlapping or exhibiting unusual poses that the model wasn't trained to handle, which leads to incorrect or imprecise keypoint predictions. The model also tends to mistake spectators for players, which could be mitigated by filtering out bleacher regions.

A major challenge was player re-identification, which is essential for triangulating keypoints across views. In fact, this is a recognized challenge within the SoccerNet dataset itself. Matching players between significantly different views is difficult, and we currently perform this step manually, assuming players can be tracked throughout the game. To automate this, future work could explore deep learning–based re-identification methods and pose embeddings.

Since no ground-truth 3D pose annotations are available, we evaluated our reconstruction qualitatively by rendering the scene from multiple virtual viewpoints. A meaningful direction for future work would be to create such a dataset by collaborating with football organizations and capturing 3D ground truth poses using motion capture systems or wearable sensors.

Lastly, deploying this system in real-world settings would face the additional challenge of synchronizing video from multiple smartphone cameras, which is critical for ensuring temporally aligned multiview frames.

5 Conclusion

In conclusion, we built a cost-effective, multi view 3D reconstruction system to facilitate soccer refereeing in amateur environments. Our system calibrated the cameras using field markers, estimated poses using an R-CNN, and triangulation to reconstruct key game moments in 3D. Even with restricted public data and noisy segmentation in certain situations, we showed that it's possible to approximate systems like VAR using just a few synchronized camera angles. This makes it easier to spot important decisions like offside and understand player positioning during plays. While we faced challenges, including not having enough synchronized camera views, bad quality images with noise, difficulty in tracking pitch markers and joint key points accurately, and needing to match players by hand, our system still showed promising results. Moreover, these limitations can be addressed with better data and improved segmentation models, and the implementation of color-based tracking, making the limitations temporary. Overall, our project shows that it's possible to make referee-assisting technology more affordable and accessible. This could help bring fairer and more consistent officiating to all levels of soccer, not just the professional leagues.

6 Contribution

Each member of our team played a vital role in ensuring the success of our project, including reading literature in the topic, identifying and setting up the datasets, building the pipeline, and writing the final report. All of us worked on the coding and writing portions and researching various tools that could support our development process.

References

Waleed Abdulla. 2017. Mask r-cnn for object detection and instance segmentation on keras and tensorflow. https://github.com/matterport/Mask_RCNN.

Shounak Desai. 2025. Real time offside detection using a single camera in soccer.

Tiziana D'Orazio, Marco Leo, Paolo Spagnolo, Pier Luigi Mazzeo, Nicola Mosca, Massimiliano Nitti, and Arcangelo Distante. 2009. An investigation into the feasibility of real-time soccer offside detection from a multiple camera system. *IEEE Transactions on Circuits and Systems for Video Technology*, 19(12):1804–1818.

FIFA. 2023. Semi-automated offside technology. https://inside.fifa.com/innovation/world-cup-2022/semi-automated-offside-technology. Accessed: 2025-03-13.

FIFA. 2025. Video assistant referee (var) standards. https://inside.fifa.com/innovation/standards/video-assistant-referee. Accessed: 2025-03-13.

Richard Hartley and Andrew Zisserman. 2003. *Multiple View Geometry in Computer Vision*, 2 edition. Cambridge University Press, USA.

Jan Held, Anthony Cioppa, Silvio Giancola, Abdullah Hamdi, Bernard Ghanem, and Marc Van Droogenbroeck. 2023. Vars: Video assistant referee system for automated soccer decision making from multiple views. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, page 5086–5097. IEEE.

Tony Lindeberg. 2012. *Scale Invariant Feature Transform*, volume 7.

Jyoti Madake, Devyani Thokal, Mohd. Ashhar Ullah, and Shripad Bhatlawande. 2023. Offside detection for better decision-making and gameplay in football. In *2023 IEEE International Conference on Blockchain and Distributed Systems Security (ICBDS)*, pages 1–7.

Floriane Magera, Thomas Hoyoux, Olivier Barnich, and Marc Van Droogenbroeck. 2024. A universal protocol to benchmark camera calibration for sports. In *IEEE International Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), CVsports*, Seattle, Washington, USA.

Karthik Muthuraman, Pranav Joshi, and Suraj Kiran Raman. 2018. Vision based dynamic offside line marker for soccer games.

SoccerNet. 2023. Soccernet: A scalable dataset for action spotting in soccer videos. https://www.soccer-net.org/. Accessed: 2025-03-13.

Author Unknown. 2025. Offside detection system using an infrared camera tracking system. *Scientific Research Publishing (SCIRP)*. Accessed: 2025-03-13.