Computer Vision Techniques for Offside Line Detection
Project Report

Zakariae EL ASRI
MSC AI at CentraleSupelec
zakariae.elasri@student-cs.fr

Nicolas GREVET
MSC AI at CentraleSupelec
nicolas.grevet@student-cs.fr

Our code is available at https://github.com/elasriz/Computer-Vision-Techniques-for-Offside-Line-Detection

Abstract

Offside detection in soccer is an important decision that can affect game conditions and, in many cases, change the outcome of the game. Offside decisions taken by on-field referees show an error rate of 20% - 26%. Before the introduction of the Video Assistant Referee (VAR) system in soccer games, a lot of games have led to controversial results due to wrong offside decisions. Currently, the offside decisions are still made by on-field referees, assisted by VAR system when the initial decision led to an error. As the offside decision is a binary problematic, VAR referees can review the supposed offside and confirm, or not, the initial decision. However, this system, in its basic configuration, has two major points of criticism: extensive delays in providing final decisions and inaccurate decisions arising from human errors. Indeed, there are some scenarios where the human can't detect accurately if it's an offside or not due to limited precision of human eye. Thus, although the paradigm of using technology to assist such decisions cannot be questioned, the inefficiencies of these methods certainly need to be addressed. This work aims to support the efforts that target these inefficiencies with the help of technology.

In this project, we will explore a computational offside detection algorithm for soccer match images that aims to aid or automate the task of making offside decision. We will use and contrast the different image processing methods like Hough transform, Canny Edge detection, and vanishing point ideas to identify the probable offside regions.

1. Introduction

According to Law 11 of the *Laws of the Game* [1] of association football, a player is in an offside position if any of their body parts, except the hands and arms, are in the opponents' half of the pitch, and closer to the opponents' goal line than both the ball and the second-last opponent (the last opponent is usually, but not necessarily, the goalkeeper). This position is not an incident, but a player so positioned when the ball is played by a team-mate can be judged guilty of an offside offence.

To detect such offence, the sideline referee tries to track continuously the last defender (exactly the second-last opponent) by moving along the sideline in the same level as this player and so to split the pitch by two areas: offside region and on-side region. When a player A is present in the offside region, the referee must be watchful to detect the moment where the ball was last touched by a team-mate B.

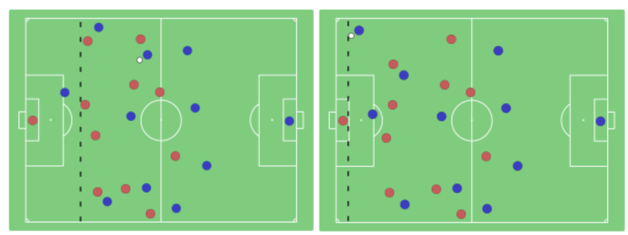

Figure 1 : ***2 configurations of the offside line marker. Blue dots are the attacking players, red ones are the defensive players, and the white spot is the ball.***

Even though these rules are very precise, the offside action is sudden in time and the referee must take decision at the split of a second. This makes the decision subject to errors of judgement usually due to the limited precision of human eye and sometimes duo to the referee inattention. As all decisions in a soccer game, an offside decision is subjective to referee. However, it is still more critical since an offside action usually results in a goal-scoring opportunity.

Traditionally, in most sports, monitoring using sensors or other devices attached to players or equipment was not generally possible. This inspired the idea of using computer vision techniques to help coaches to analyze the performance of opponents and improve/adapt the performance of their team, as well as for media professionals to make game analyses for the public. Recently, these techniques are also being used to assist referees in their decisions in soccer, as so in many other sports. *Thomas et al.* [17] presented in 2017 a survey of some applications that use computer vision for sports analysis, some of the topics that were being addressed to the research community.

CV techniques can be applied to many scenarios in a soccer game, especially for scenes of possible offside that is difficult to judge for the referees and even critical for the outcome of a game. In this work, we present a pipeline of computer vision techniques that would be able to help referees to make less errors when judging an offside. We specifically determine the marker line splitting the play area in two sub-areas (on-side and offside) from footages where the images have been collected in such a way that they present a diverse collection of events that can occur in a game. These include various scenarios from both sides of the field.

2. Related Work

From the offside's definition, the solution for this problem must include several tasks. Indeed, we first have to detect the ball and people on the pitch, then classify them by attackers, defenders, referee, and goalkeeper. The determination of the second last defender to define the offside region come in a second part. To manage this, many researchers have done several works to develop some or all of tasks above.

To detect and track player positions, many researchers thought about using multiple cameras: *M. Xu et al.* [18] wield the use of eight static cameras when they describe a system for tracking the positions of football players during a match using object detection and tracking, first in the image plane with single camera, and then in the ground plane fusing measurements from each camera. T. *D'Orazio et al.* [2] studied later the feasibility of real-time soccer offside detection from six static cameras placed on the two sides of the soccer field. This technique was used on another context by *T. Saba et al* [14] to detect offside event but from a semantic point of view such as video summarization, features analysis, and provision of augmented information. *S. Hashimoto et al.* [5] have worked to make offside detection by calculating 3D texture from multiple fixed cameras.

W. L. Lu, et al. [8] thought about the use of a single pan-tilt-zoom camera to track and identify players in sports videos. They apply a tracking-by-detection approach to track sports players in video streams. Specifically, they run the player detector called Deformable Part Model (DPM) [3] to locate players in every frame, and then they associate detections over frames with player tracks using a one-pass approach.

Other researchers worked on a single image or on multiple frames of a video. *Galaviz et al.* [4] employ image stitching techniques across multiple frames of a video for player tracking. After detecting players and extracting backgrounds, they calculate a homograph with a reference image to uniquely determine the player position in the given video input. Then, they perform a Real Time Player Tracking by differentiating between background pixel values and frame pixel values

Some unusual situations in soccer games are confusing, for example sometimes the goalkeeper is not the last player from his team, so if we will be wrong if we track the last defender to make offside decision. To address this problem, *J. R. Nunez et al.* [11] describe a methodology for player/referee segmentation in soccer games. They developed tools for detecting dominant color regions to detect referee and players then they performed an unsupervised clustering for segmentation task.

Many other researchers used an unsupervised algorithm to perform players classification *P. Spagnolo et al.* [16] and *P. L. Mazzeo et al.* [9].

The previous works don't capture all aspects of the offside rule. Indeed, "… a player is in an offside position if any of their body parts, except the hands and arms, are in the opponents' half of the pitch …", so the tracking players is not accurate for offside detection, and we must track only the playable parts of his body (don't include hands and arms). To address this issue, *N. Panse et al.* [12] present a computational offside decision algorithm for soccer match images where they use pose estimation to identify the farthest horizontal projection of the playable body parts of all players, within a pipeline of Computer Vision tasks, that include also calculating the vanishing point to determine the relative position of all the players with respect to the field. Then, they classify players and referee by clustering their jersey colors, then highlight players in the offside region.

All the methods above are good for a semi-automation offside detection. Where a referee must first track himself the time where a ball pass had occurred, and then use the system to verify if the pass receiver was in offside position. To build a system for fully automated offside detection, ball tracking is mandatory. But this task is not that easy as the ball is a fast-moving object and so it gives a blurry image from video.

For ball tracking, *J. Hossein-Khani et al.* [6] describe a framework to detect the ball on the field. They separate first foreground and background part from the captured images and convert it into binary images by using a thresholds method.

P. Khirwadkar et al. [7] propose two models to handle the problem of offside detection. The first model would take care of detecting the ball, tracking it, and detecting whether a ball pass has occurred. The second would detect the players of each team, attacking and defending, and get an approximate location of the players. Then, they combine ball tracking and the players tracking models in one single program that detect offside actions in scenarios of two attackers and one defender.

Since deep learning had shown tremendous results during the last years on image segmentation, *J. Singh et al* [15] thought to build a system which integrates Computer Vision techniques and Deep Learning for offside detection. They first apply a convolutional neural network on a static image for detecting players. Then they use computer vision techniques like dominant color extraction to detect features that will serve for a clustering method to assign each player to its team.

In sports events generally and especially soccer, the shadow is one of the most complex barriers to track players especially when a game occurs at night. Indeed, it's falsely classified as foreground. To tackle this issue, *J. Renno et al.* [13] propose an unsupervised learning procedure that determines the RGB color distributions of the foreground and shadow classes. They used a filtering process and pixel classification mechanism to distinguish the foreground and shadow.

As part of our academic project, we were inspired by the work of *K. Muthuraman et al.* [10] where they describe the process of drawing a dynamic offside line with computer vision techniques. They first determine the play area using a Hough transformation, then they use color similarity to identify regions with similar color profiles and to identify players. Later, they track players with powerful tracking techniques and determine the vanishing point by exploiting the existing lines on the image.

3. Methodology

In this project, we handle the problem of offside detection by proposing a complete pipeline of computer vision tasks. Due to time limitation, we were not able to tackle all the challenges for the offside detection. So, we underline some assumptions on the offside rule to simplify its treatment and to focus more on some interesting aspects regarding the computer vision methodology.

Our approach is based on the idea that it is not a big challenge to semi-automatized the process of creating an assistant tool detecting possible offside on an image. This can be done with the aid of a marker line. The referee gives to the system an image selected among a set of footages, and he gets the same image with the offside line highlighted as output. The algorithm can then determine if one or multiple attacking players are offside or not.

The data collection process was easy because it is frequent to see soccer games in live or in replay nowadays so we just collect some screenshots of games at any time of the game where a diverse collection of events can occur, on both sides of the field. Initially, our dataset had 492 images from 10 different games, so with approximatively 50 images for each game. But we had some issues when trying to automatize the preprocessing part for each image, so we took the decision to restrain our dataset to keep footages from one game and from one single camera fixed on the middle side and for situations on the right part of the field, we perform a rotation along the vertical axis of the image to as if it was on the left part. These assumptions allow to simplify the parametrization of our algorithm regarding the values of different masks applications for example. Our final dataset consists in 48 images of the Euro 2016 game: Poland vs. Portugal.

Figure 2 : ***First original image of our dataset. Poland players are in white (goalkeeper in yellow), Portugal players are in red (goalkeeper in black). Referees have sky-blue jerseys and black shorts.***

These following points are considered as prior knowledge for the further described pipeline:

1. The goalkeeper is always the last player, and he is not considered when drawing the offside line; the detection of second last player will be our main objective.
2. Relaxation of the offside rule: a player is offside if any of his body part is in the offside region. Indeed, to perform pose estimation at each player level, and so to detect every body part, we made further research without finding any solutions that did not apply Deep Learning techniques, which was a major constraint to respect on this project.
3. No detection of the ball as it is a semi-automated project, the user must visually identify the attacking and defending teams by the colors of their jersey.

The user must give to the algorithm the following parameters as inputs:

- the jersey's color for attacking/defending teams
- if the image needs a rotation, so as the action is always happening to the left direction
- a threshold to adjust the number of pixels considered when classifying the players

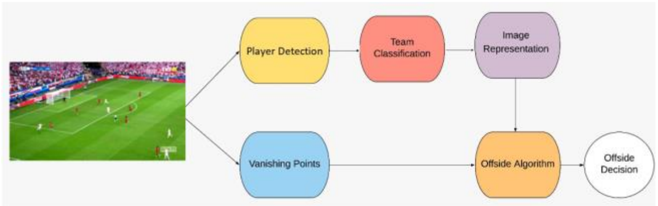

Figure 3 : ***The complete pipeline of our system***

3.1. Vanishing points

The most crucial and important step when detecting an offside is to compute the relative positions of players in the 2D play-area. For this, we must counteract the distortions introduced by camera placements in the side-view. The best way to do this is to find the vanishing points. For this task, we use the marking lines existing on the field.

To help the user to extract the field lines, a control panel is implemented to manually adjust the values of Hue, Saturation and Values. When the good HSV values are set on the control panel, they are automatically added on the system as global parameters. This part is a color-based filtering.

The algorithm is applied on several step of our process to segment regions of interest rigorously. We start each segmentation by first converting RGB images into the HSV space that has shown to be more suitable for this task. Given an interval of Hue, corresponding to the color of interest, as input. The color-based filtering begins with Hue component threshold that consists in binarizing Hue component from an interval [H_{min}, H_{max}].

All pixels with Hue value into this range are set to preserves their original color (the region of interest) and the rest is set to black. This process permits to create a binary image where regions of interest are highlighted and surrounded by a black background.

Input images may contain some noise patterns such as substitutes, audience, publicity spots, people outside the play-area … To address this point, it is crucial to determine the pitch. Since the camera is fixed, the images will be taken with different angle. So, the system must be stable to the varying translation. Exploitation of the field color (green) and application of the color-based filtering to identify regions with similar color profiles are the solutions adopted. Extraction of only the significant connected components is done and the play-area is the largest connected component. This is helpful if we want to discard all the detected patterns outside this area which means removing the audience, for example.

Then to detect edges, the Canny Edge detection algorithm is performed and the Hough Line algorithm to extract lines from the image. The lines are then extended, the pairwise intersections computed and averaged out, this is the vanishing point. This process is done for vertical and horizontal lines separately. This will be helpful to compare the relative distance between each player and the goal line, by comparing the angle between the lines connecting a player to the two vanishing points (horizontal and vertical).

3.2. Player detection

First step towards successful offside detection is the accurate localization of players. Each player needs to be detected and classified into specific categories such as offside, onside, last defending player or defending player. As observed in the review previous work, simply detecting players as bounding boxes does not prove useful in determining the farthest line of the player as the boxes may cover more area than the actual body of the player. Moreover, the offside rule checks for the farthest allowed body part of the player which cannot be represented accurately by the four coordinates of the bounding box and the pose estimation is often preferred for such tasks

Nevertheless, as explained in introduction of the methodology part, during the development phase of this project, there is no pose estimation techniques that does not use Deep Learning methods. So, we decided to use the bounding boxes around players.

As the input frames also contain unwanted detections such as substitutes, ball boys and people in the audience, it is necessary to segment the playable area of the field. This is done in two steps. First, a color mask is applied to the image to segment the grass field from the image. Secondly, morphological transformations are applied to the masked image to segment the audience from the playable field area and to preserve homogenous regions for players without changing their geometry. Any predictions outside the field under consideration are discarded.

We must note that the jersey of each player is not unicolor, indeed, there must be player identification numbers, names, and sponsor logos. This will result in a noise within regions detected as representing players. To address this point, we apply an image filling operation wherein we fill holes to represent players as continuous regions. This filtering process preserves and affine the regions of interest.

3.3. Team classification

In the last process, we preserved regions that could be candidate to represent a player. We apply a first a mask filter using the play-area, previously identified, to eliminate regions and patterns outside the play-area. Then we exploit

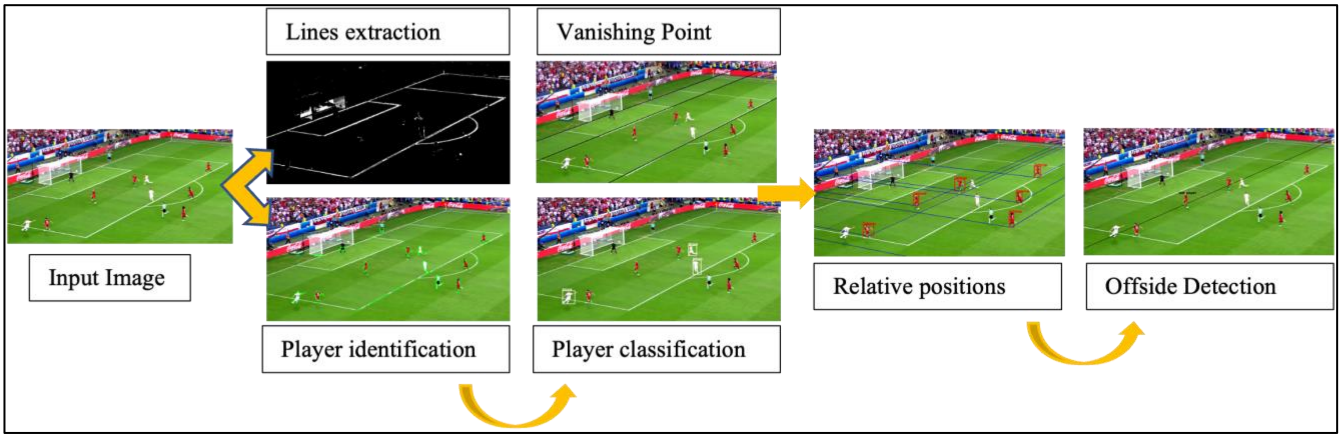

Figure 4 : ***The step-by-step process of offside detection illustrated on the original image of the dataset***

the information of the team jerseys (given as input), to perform the color-based filtering on the remaining regions to extract only the ones satisfying the thresholds condition. Hence, we highlight all the players, and more precisely the players wearing jersey respecting the team color (except the goalkeeper since he has a different color of jersey).

After applying this first filtering, it may remain some noisy regions within the play-area, that don't correspond on any player. This might be a result of lighting noise or color variations. To eliminate this additional noise, we will count the number of pixels, within each region, that satisfy the jersey color similarity, then we will preserve only regions where this number is higher than a given threshold.

Now, we preserved only players of each team, we will assign a bounding box for each one of them to represent well the results.

3.4. Drawing the offside line

Once the vanishing point determined and the player bounding boxes identified, the last step is to find the last defender (*i.e* the closest defending player to the goal). For this, we need to compare the real distance between each defending player and the goal line. We extend a line from each defender to the vanishing point to and then we select that line which has the lowest y intercept on the left of the image.

<u>*Proof:*</u>

Let A and B two players in the play-area.

Since V is the vanishing point, then the distances D_a and D_b are respectively proportional to angles $\boldsymbol{\alpha}$ and β.

So,

$$D_a < D_b \Leftrightarrow \alpha < \beta$$
$$\Leftrightarrow \alpha + \gamma < \beta + \gamma$$

Since $\boldsymbol{\alpha}+\boldsymbol{\gamma}$ and $\beta+\boldsymbol{\gamma}$ are $\leq \frac{\pi}{2}$ (because we assume that we study the left side)

$$\boldsymbol{D_a} < \boldsymbol{D_b} \Leftrightarrow \tan(\alpha + \gamma) < \tan(\beta + \gamma)$$
$$\Leftrightarrow \frac{Y_A}{OV} < \frac{Y_B}{OV}$$
$$\Leftrightarrow \boldsymbol{Y_A} < \boldsymbol{Y_B}$$

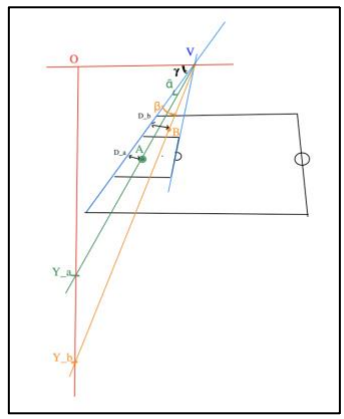

Figure 5 : ***Mathematical representation of the offside decision***

3.5. Making offside decision

With the offside line calculated in the previous step, we can now detect if an attacker is in the offside region or not. All attacker having the intercept between the line connecting him to the vanishing point and the Y-axis in the left of the image lower than the corresponding intercept for the defending player will termed as offside

$$Attacker\ is\ offside \Leftrightarrow \boldsymbol{Y_{Attacker}} < \boldsymbol{Y_{Last\ defender}}$$

The corresponding algorithm is described by:

```
Input: Players, A_Team_Id, D_Team_Id
Output: Offside Decision
// find The last Defender
D_Team_Intercept = []
for Player in Players do
.. if Player[id] == D_Team_Id then
.... D_Team
Intercept.append(YinterceptV])
Sort(DefendingTeamIntercept, Ascending)
MinIntercept = DefendingTeamIntercept[1]
// Check attacking team players for
offside
for Player in Players do
.. if Player[id] == AttackTeamId then
.... if Player[YinterceptV]<Minintercept
then
........ Player[OffsideDecision] = True
........ DrawBoundigBox(player)
........ DrawOffsideLine
```

4. Experiment

4.1. Metrics

An objective evaluation methodology requires the availability of ground-truth data measures a quantitative performance of the proposed methods. In addition, the gap between estimations and ground-truth provides interesting insights for making corrections and optimization of the baseline method. However, we notice lack of available labeled datasets in the soccer field, especially for offside detection.
Unfortunately, the dataset we used in this project [12] was labeled with the positions and geometry of players. Although, the most important for us is labels corresponding to the offside decision.
To evaluate our system, we generate a small subset (20 images) from the images and manually completed it by adding offside labels (last defender position) to represent the ground truth for the evaluation.

In our experiment, the system is a sequence of several tasks. So, the errors in some earlier tasks will have an impact in the performance of the next ones and so on the final system output. Hence, we will individually evaluate each task.

- Evaluation of Player identification and classification:
This task is a pure classification. Since a misidentification or misclassification (either False positive or False negative) of one player can drastically impact the offside decision, the appropriate metric for this task is the F1 Score for the evaluation.

- Evaluation of last defender relative position:
This task is the most crucial, and it's the main task for our system. In this step, we can think about using a metric that calculate the gap between the estimated position of the last player and the ground truth position. The MSE (Mean Squared Error) seems to be an appropriate metric for this system.

4.2. Results

F1 Score		Offside Line Error
Defending Team	Attacking Team	
69%	77%	0.1

We performed several experiments with different initialization values for the parameters (Hue range, angle for vanishing …)

The result calculated allowed us to optimize our system. We noticed that for most examples, we were able to achieve a perfect score on the classification task (F1-Score ~ 1) and a small error on the line offside estimation task.

However, for some examples, the score of classification weren't good enough, and this caused in certain case the wrong identification of the last defender with an error quite high.

5. Conclusion

In this work, we propose a semi-automated system for offside detection. It could be used as a support for the VAR referees during the game. The system is a fully Computer Vision framework, it is interactive and allows the user to give his prior knowledge as input parameters and to adjust some parameters later for better results.

Our main objective was not to look for a system with irrelevant accuracy, but to discover and exploit a maximum of interesting techniques from computer vision to build a baseline system with good accuracy. The proposed system can achieve better result with some improvements that we proposed in the discussion part.

The experiment shown good results on identification and classification part in addition to the relative position estimating part. We must notice that we tackled this problem under several assumptions that simplified it, but it still a good baseline to start with for future works and try to handle the other challenges of the offside rule.

In addition to the good result, the system proposed has shown some limitations in certain situations that we discuss in this part.

- Failure of player classification due to Overlapping: When representations of two or more players overlap in the image, the system considers the players as only one player.
- Failure of player detection due to occlusion: When the overlapping is total (ie when a player is hidden on the image by another player), it's evident that this player is ignored by the system.

An efficient solution for the overlapping and occlusion problems is to use images simultaneously from multiple cameras well placed around the pitch.

- Wrong identification of the last defender: For the relative position estimation, the vanishing technique gives reliable results. However, for the player representation, we took the central point and connected it to the vanishing point.

For rigorous result, we can perform orthogonal projection of all the body parts on the field, then connect them all to the vanishing point and determine the nearest point to the goal line.

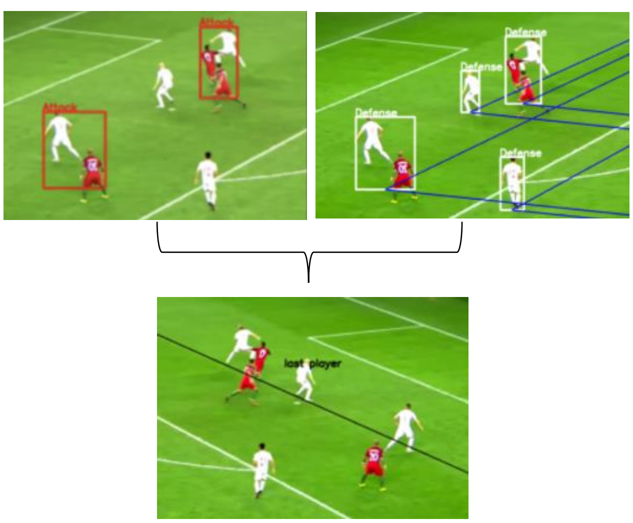

Figure 6 : ***Bad representation of the offside line due to overlapping bounding boxes***

6. References

[1] "Law 11 - Offside". *Laws of the game of Association Football*.https://www.theifab.com/laws/latest/offside/#offside-offence .

[2] T. D'Orazio, M. Leo , P. Spagnolo , P. L. Mazzeo , N. Mosca, M. Nitti , and A. Distante. An investigation into the feasibility of real-time soccer offside detection from a multiple camera system. IEEE Transaction on Circuits And Systems for Video Technology 1804–1818. Beijing. 2009

[3] P. Felzenszwalb, D. McAllester, and D. Ramanan; A Discriminatively Trained, Multiscale, Deformable Part Model. in CVPR, 2008.

[4] A. Galaviz et al. https://github.com/AndresGalaviz/Football-Player-Tracking

[5] S. Hashimoto and S. Ozawa. A System for Automatic Judgment of Offsides in Soccer Games. IEEE International Conference on Multimedia and Expo, 2006.

[6] J. Hossein-Khani, H. Soltanian-Zadeh, M. Kamarei and O. Staadt. Ball Detection with the Aim of Corner Event Detection in Soccer Video. IEEE Ninth International Symposium on Parallel and Distributed Processing With Applications Workshops. 2011.

[7] P. Khirwadkar, S. Yenamandra, R. Chanduka, and I. Juneja https://github.com/kparth98/ITSP-Project

[8] W. L. Lu, J.A Ting, J. J. Little, and K. P. Murphy. Learning to Track and Identify Players from Broadcast Sports Videos. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 35, no. 7, pp. 1704-1716, 2013.

[9] P. L. Mazzeo, P. Spagnolo, M. Leo, and T. D'Orazio, T. (2010). Football Players Classification in a Multicamera Environment. Advanced Concepts For Intelligent Vision Systems, 143-154. 2010

[10] K. Muthuraman, P. Joshi, and S. K. Raman. Vision Based Dynamic Offside Line Marker for Soccee Games. arXiv:1804.06438. 2018.

[11] J. R. Nunez, J. Facon, and A. Souza. Soccer Video Segmentation: referee and player detection. 2008 15th International Conference onSystems, Signals and Image Processing. vol. 1, pp. 279–282, IEEE. 2008.

[12] N. Panse, and A. Mahabaleshwarkar. A Dataset & Methodology for Computer Vision based Offside Detection in Soccer. MMSports'20, Seattle, WA, USA. 2020.

[13] J. Renno, J. Orwell, D.Thirde, and G.A. Jones. Shadow Classification and Evaluation for Soccer Player Detection. Procedings of the British Machine Vision Conference 2004.

[14] T. Saba, and A. Altameem. ANALYSIS OF VISION BASED SYSTEMS TO DETECT REAL TIME GOAL EVENTS IN SOCCER VIDEOS, Applied Artificial Intelligence, 27:7, 656-667, 2013.

[15] J. Singh https://github.com/jagjeet-singh/Vision-based-method-for-offside-detection-in-soccer

[16] P. Spagnolo, P. L. Mazzeo, M. Leo, and T. D'Orazio. Unsupervised Algorithms for Segmentation and Clustering Applied to Soccer Players Classification. SIGMAP (pp. 129-134). 2007.

[17] G. Thoma, R. Gade, T. Moeslund, and A. Hilton, Computer Vision for Sports: Current applications and research topics. Comput. Vis. Image Underst. 159, 3–18. 2017.

[18] M. Xu, J. O. and G. Jones. Tracking football players with multiple cameras. Image Processing. ICIP International Conference on, pp. 2909-2912 Vol. 5. 2004.