Automated Offside Detection by Spatio-Temporal Analysis of Football Videos

Ikuma Uchida*
Atom Scott*
University of Tsukuba
Tsukuba, Ibaraki, Japan
uchida.ikuma@image.iit.tsukuba.ac.jp
scott.atom.ta@alumni.tsukuba.ac.jp

Hidehiko Shishido
University of Tsukuba
Tsukuba, Ibaraki, Japan
shishido@ccs.tsukuba.ac.jp

Yoshinari Kameda
University of Tsukuba
Tsukuba, Ibaraki, Japan
kameda@ccs.tsukuba.ac.jp

ABSTRACT

In this paper, we propose a new automated method to detect offsides from football match videos. The advantage of our method is that it can strictly follow the official offside rules in which the dynamics of play actions are spatio-temporally investigated. Furthermore, to overcome the difficult task of tracking the two-dimensional locations of the players and the ball, we utilized geometric characteristics on the perspective projection coupled with a Kalman filter to estimate information necessary for offside detection. Based on these methods, our prototype system can recognize whether an attacking player who crossed the offside line receives a pass from their teammate or not. To the best of our knowledge, our proposed method is the first method that can automatically determine offsides from video. Furthermore, this method is designed to enable online processing in the future.

CCS CONCEPTS

• **Computer systems organization** → **Pipeline computing**; *Complex instruction set computing*; *Special purpose systems*.

KEYWORDS

offside, object detection, tracking, team classification

ACM Reference Format:

Ikuma Uchida, Atom Scott, Hidehiko Shishido, and Yoshinari Kameda. 2021. Automated Offside Detection by Spatio-Temporal Analysis of Football Videos. In *Proceedings of the 4th International Workshop on Multimedia Content Analysis in Sports (MMSports '21)*, October 20, 2021, Virtual Event, China. ACM, New York, NY, USA, 8 pages. <https://doi.org/10.1145/3475722.3482796>

1 INTRODUCTION

The offside rules of the International Football Association Board (IFAB) states that an attacking player is offside if, at the moment of the attacker's pass, any part of the head, body, or feet is in the opponents' half (excluding the halfway line), and any part of the head, body or feet is nearer to the opponents' goal line than both the ball and the second-last opponent [1]. The refereeing of offside

*Both authors contributed equally to this research.

This work is licensed under a Creative Commons Attribution International 4.0 License.

MMSports '21, October 20, 2021, Virtual Event, China
© 2021 Copyright is held by the owner/author(s).
ACM ISBN 978-1-4503-8670-8/21/10.
<https://doi.org/10.1145/3475722.3482796>

Calibrated Soccer Videos

Figure 1: Overview. We automatically detect offside by tracking the players and the ball in a match video and classifying each team.

is a complex process and requires split-second decision-making. Traditionally, an offside decision has been made by the assistant referee moving along the court's sideline and following the second last defender (also known as the offside line) of a particular team. This method of refereeing places a heavy physical burden on the assistant referee during the game. Furthermore, it is inconsiderate to the limitations of human vision [2]. For this reason, offside is one of the most common fouls in football, and is relatively prone to misjudgment [2–4]. There is also a 20%–26% error in offside decisions, depending on the referee [2, 5]. Therefore, it is essential to provide technical support to ensure consistency in the judgment of offside.

The past decade has seen the proliferation of technology-enhanced refereeing. For example, video assistant referee (VAR) systems have been added to the technology-assisted decision-making toolkit in recent years [6, 7]. However, VAR ultimately performed by a visual inspection of video footage. Therefore it is nonetheless a subjective decision made by the referee [6, 8, 9].

There have been many studies on the automatic detection of offside scenes by computers [10–16]. These studies have contributed greatly to the quantitative judgment of offsides. However, offside rules need to be determined by taking into account play events obtained from the temporal information in the video sequence, such as passes and ball touches. These studies do not take such dynamic

event information into account. Therefore, no effort has been made to automatically judge offside as defined by the IFAB.

This study aims to quantitatively express a more rigorous definition of offside by including temporal information. We propose a computer vision pipeline to automatically identify players judged to be offside from the match video. In the offside rule, the offside player must be not only in the offside position at the moment of the pass but also involved in the pass. In order to identify the players involved in the pass in a video sequence, it is necessary to track the positions of the each player as they move from frame to frame. In our proposed pipeline, we first detect the positions of the players and the ball. Then the team and ID of a specific player are assigned in each frame. Finally, passes are automatically detected and offsides are determined. By incorporating an offside detection algorithm that considers the detected pass information, we can represent a tough offsides decision in a computer. In the case of broadcast images, there are many cases where the player who is deemed to be offside by the referee is not included in the image. Therefore, considering that professional and amateur football teams always record videos in which all the players in the field are visible, we use as input the videos recorded by two cameras so that all the players are visible in the angle of view.

The contributions of this study are as follows.

- To the best of our knowledge, this is the first automatic offside detection approach that takes the dynamics of play into account to implement the IFAB offside rules precisely.
- We propose a novel pipeline to track players via kinematic Kalman Filters and assign team IDs by solving a constrained general assignment problem based on dominant colors.
- We demonstrate a method of detecting offsides from video using inexpensive equipment. This method can benefit amateur teams that have tight budgets as well as well-funded professionals.

In the next section, we give an overview of previous work on offside detection. In this section, we mention the different computer vision tasks used and what role they play in the overall process.

2 RELATED WORKS

Several studies have been conducted to analyse the problems that occur when assistant referees make offside decisions and to improve the accuracy of these decisions. [2–5]. These efforts have revealed the common problems in refereeing, such as occlusion from the assistant referee’s point of view and human errors due to visual illusions. The introduction of the VAR system has shown an improvement in accuracy by using multiple camera views and replays to tackle these problems. On the other hand, the intervention of the VAR requires a delay of 5 or 6 minutes in some matches to confirm the decision [17]. The accuracy of the VAR review of offside decisions is also called into question due to errors in the offside lines drawn manually by the VAR. In light of this and understanding the difficulties of making offside decisions, several attempts have been made to automate offside decisions.

The use of video feeds from multiple cameras has been used to tackle the problem of obstructed views on the pitch [11, 12, 18]. However, these studies do not provide sufficient detail on how to perform the necessary sub-tasks on the image data, such as finding the location of a player’s body part, tracking a player with the ball, or classifying players into teams. K. Muthuraman. has used computer vision and image processing techniques to mark the offside line [15] dynamically. This study provides a method of tracking players in successive frames of a football match video using the Kanade-Lucas-Tomasi tracker and acquiring and using vanishing points to determine the relative positions of the players. On the other hand, the positional relationship between the ball and the players has not been verified. Neeraj. et al. provide an approach to accurately identify a player as offside by using posture estimation and unsupervised team classification techniques [10]. However, it does not provide temporal event detection, making it difficult to deal with offside scenes that change depending on events during the match. Parth et al. use ball tracking and player tracking systems to detect passes and use relative positions to determine offside [19]. On the other hand, the use of the system on a regular pitch size is limited as the detector is only compatible with some match spaces. Also, the problem of player occlusion in the image is not discussed. With the availability of large broadcast video datasets, including offside scenes, methods for action spotting using Spatio-temporal features from the broadcast video are also available [20–22]. Giancola. et al. incorporate a pooling module into the action detector that independently considers the context before and after the action occurs [21]. Cioppa. et al. detect scenes by focusing on the feature representation of each frame and the graph structure of the players’ positions. They also describe a calibration method for projecting the player positions from the broadcast video onto the top pew image [22]. On the other hand, these action spotting methods also consider the zoomed images of the players and the out-of-play image features to estimate the context of the event. Therefore, it is not easy to represent these features sufficiently in the available video, leading to false detection of events.

The task of automatic offside identification consists of various sub-tasks such as player and ball detection, team classification, etc. The offside decision is then made by detecting the attacking player’s pass, which is obtained from the output of these tasks. For object detection, standard object detection techniques such as [23] and [24] are used, as well as [25, 26], which are also used in sports analytics. In the task of team classification, clustering-based approaches can be used to automate the process. Neeraj. et al. use a DBSCAN classification method for the color information of teams [10]. Koshkina. et al. propose a deep learning-based classification method using contrastive learning [27]. On the other hand, to the best of our knowledge, an effective method to classify goalkeepers by clustering and assign them to specific teams has not been solved. Theagarajan et al. have implemented a feature matching algorithm using prior information about the two teams and the color of the goalkeepers’ uniforms [28]. In terms of pass detection during a match, several attempts have been made to detect passes from broadcast video. Sorano et al. detect the bounding box of the player and the ball and use LSTM to detect pass events [29]. Sarkar et al. automatically identify valid passes in a match by using a minimum cost flow network. While these are robust pass detection models

Figure 2: The proposed pipeline of offside detection. We detect the players and the ball in the calibrated match video. Next, we focus on the ball trajectories and velocities to identify the frames where the passes occurred. We then track the players and classify the teams among the frames. In this way, we can detect the offside efficiently when a passing event occurs.

using match footage, they often misclassify passes and dribbles. In addition, it is not easy to detect a pass when the passer and receiver are not in the same frame, such as a long pass. Our approach is to use multiple wide-area videos. This ensures that the passer and receiver are always in the same frame. We also distinguish between passes and other ball touches based on information about the speed of the ball’s movement.

3 METHOD

3.1 Overview

This work presents a method for automatically detecting offsides in football match videos as a pipeline in a specific order. In this section, we describe each of the tasks and their role in the overall process. The pipeline for automatic offside detection is shown in Fig. 2.

The first step is to cancel out the effects of camera distortions in order to detect the exact positions of the players and the ball. Next, we detect the objects in the calibrated video and project them onto the pitch plane. We consider that the path events occur between the frames corresponding to the extremes of the ball velocity in the pitch plane. This study focuses on the ball’s velocity in the pitch plane and identifies the frames corresponding to the maximum and minimum values. We then classify the teams and track the players between these frames. In this way, we can efficiently identify the passer and receiver of a pass. Finally, based on the information

obtained from these subtasks, the core offside algorithm makes the final offside decision.

3.2 Calibration

The offside detection requires an accurate understanding of the relative positions of the players and the ball on the field. We, therefore, compensate for distortions caused by camera placement and lens angle. Camera calibration allows us to know the part of the field recorded by the camera and the relative positions of the player and the ball on the field. In the work of K. Muthuraman. and Neeraj, The concept of vanishing point is used to determine the relative positions of the player and the ball. In this way, it is possible to calculate the object’s position, taking into account the distortion of the camera. On the other hand, when multiple cameras are used, it is difficult to identify the scene’s exact distortion, which is commonly observed in the camera images. In this study, we use Zhang’s method to calculate the distortion parameters of the two fixed cameras used [30]. As shown in Fig. 3(b), we use Harris’s corner detection method to detect the feature points of the chessboard taken from several angles. Then, we calculate the internal and external parameters of the fixed camera and the distortion parameters. The distortion of the camera image is then corrected based on the calculated camera parameters (Fig. 3(c)).

3.3 Player and Ball Localization

In order to successfully detect offside, we need to locate the exact positions of the players and the ball. In this paper, we propose a

Figure 3: Construction of the input frames. (a) An example of two camera frames. (b) Several pictures of the chessboard taken by the cameras and the camera parameters and distortion parameters are obtained from these feature points. (c) Example of a frame calibrated with the obtained parameters.

new method to identify the players' positions and the ball at high speed as the match video is updated. We detect each player and the ball and obtain the bounding boxes of the player and the ball from the 2D frames obtained from the fixed camera.

You Only Look Once version 4 (YOLO) Object Detection enables the detection of balls and players by scanning the image once [23]. YOLO v4 detects, from the input image, the object's position in the image coordinates, the confidence level (accuracy of the prediction model), and the number of instances. YOLO v4 then adds a bounding box around the detected object. The confidence is defined by the predicted value and the actual value, as shown in Eq. (3.1). If no object is detected, the confidence score is set to zero. The probability that an object is present in a grid cell is defined as $Pr(Object)$.

$$\text{Confidence} = Pr(Object) \times \frac{\text{Area of overlap}}{\text{Area of union}}. \quad (3.1)$$

In this work, we use YOLO v4 with enhanced mosaic data implemented in Pytorch known as YOLO-v5 [31].

In this study, we focus on the fact that the camera motion is fixed, and identify the keypoints of the pitch in the image by annotation. Based on the location of the keypoints, we obtain the homography matrix and project the obtained position coordinates onto the pitch plane. When the image plane is projected onto the pitch plane, the

Figure 4: Time series of ball trajectory and ball velocity. (a) Example plot of ball detection results. (b) An example of the reconstructed trajectory of a ball after setting a threshold value for the ball velocity and applying a smoothing process. The colour information in the detected bounding box is used to optimally assign a team.

object's bounding box is elongated linearly concerning the camera's focal point. In order to avoid false projections onto the pitch plane due to this elongation, we refer to the lower-right coordinate of the bounding box and project it onto the pitch plane. In this way, the frame-by-frame position of the offside player and the ball and the variation of the offside line can be visually grasped from the pitch plane.

3.4 Pass Situation Detection

In the offside rule, A player in an offside position is penalized for being involved in a pass. Therefore, detecting the frame in which the passing event takes place is essential for automatic offside detection. In this study, we classify and track the teams between the frames in which a pass is made.

Xinguo tracks the ball velocity over time and focuses on its extremes in order to detect the frame in which a pass is made [32]. In this paper, the ball is assumed to be in the pitch plane, and the ball's position in the pitch plane is determined. The maxima value of the ball velocity is detected as the frame in which the pass is made. The minima value of the ball velocity is detected as the frame in which the pass is received.

Regarding the ball velocity, Fig. 4(a), we observe that when the ball flies through the air in 3D space, there are frames where the apparent ball velocity is exceptionally high. Between frames with such exceptionally high velocities, the ball's flight pattern is affected by the projective geometry. Therefore, the projected trajectory of the ball differs significantly from the actual trajectory of the ball in the pitch plane (Fig. 4(a)). This study identifies the path event based on the distance between the player and the ball. Therefore, distortions of the ball trajectory may lead to false detection of the frame where the pass event occurred. In order to prevent such false detections, we use the velocity of a football ball kicked out in real space as a reference and set a threshold to remove the reference to the section where the ball velocity is exceptionally high. Then,

Figure 5: A team classification method based on GAP. Based on the colour information in the detected bounding box, it assigns teams optimally.

the ball velocity in the removed section is corrected by linear interpolation. As shown by the red line in Fig. 4(b), the ball motion is correctly approximated after linear interpolation. We notice a large amount of noise is detected in the high-frequency component of the ball velocity. The noise may be due to the bouncing of the ball in 3D space. In this study, the effect of the noise is reduced by calculating the moving average of the ball velocity.

3.5 Team Classification

Team classification is an important task to identify the teams to which the passer and receiver belong. Therefore, we perform team classification among the frames obtained by the sub-tasks described above. This study classifies the teams by focusing on each frame's color information in the bounding box region. An overview of the team classification task is shown in Fig. 5.

We quantitatively classify the color information contained in the bounding box obtained from the object detection task as five different color palettes[33]. Here, we obtain the dominant color in the color palette. In this study, after classifying the color information, we prepare five color templates (team A, team B, goalkeeper of team A, goalkeeper referee of team B) and calculate the color difference between the color information in the color palette and the color templates. The color difference between one of the color palettes

Figure 6: Comparing the predicted trajectory of a player by KF with the observed trajectory. The predicted value is generated from the observed value, and the state of the player is corrected by the predicted value of KF in case of detection failure.

and one of the color templates for a given player is calculated.

$$CD = \sqrt{(R_{CP} - R_{CT})^2 + (G_{CP} - G_{CT})^2 + (B_{CP} - B_{CT})^2}.$$

R, G, B : Values of Red, Green, Blue

CP : Color Pallet

CT : Color Template

(3.2)

The color difference is calculated as the Euclidean distance in the RGB color system. Thus, the dominant color is the color that minimizes the color difference. Next, we assign the dominant color to the location information of each team. In this study, we optimally solve the General Assignment Problem (GAP) such that the CD assigned to each team is minimized, as shown in Eq. (3.3) [34], and assigns team IDs to the detected players.

$$\begin{aligned} \text{minimize } & \sum_{i=1}^m \sum_{j=1}^n CD_{ij} x_{ij} \\ & x_{ij} \in 0, 1. \end{aligned} \quad (3.3)$$

The m corresponds to the number of people in the pitch plane ($= 25$) and n to the type of team colors ($= 5$). The x satisfies the property Eq. (3.3). For the constraint of GAP, we require that the total number of members in a football court is 25, including 20 field players (10 for each team), three referees, and two goalkeepers.

3.6 Player and Ball Tracking

Based on the positional coordinates of the players and the ball in the pitch plane detected from the match video, we assign the players' positions and the ball for each frame. Then, we dynamically detect the defender who is the reference of the offside line and identify the attacking player who made the pass. In this work, we introduce the concept of Simple Online and Real-time Tracking

Figure 7: A method for detecting pass scenes based on the ball velocity. First, we detect the frame in which the ball velocity reaches its extreme value. Next, we identify the player who is closest to the ball as the player involved in touching the ball. Then, we detect the pass of the attacking team based on the information of the player’s team obtained beforehand.

(SORT) technology to perform multi-objective tracking and fast frame-to-frame mapping of the players and the ball [35].

In order to generate estimates of player and ball positions in the presence of noise, detection failures, and outliers due to object detection, the Kalman filter (KF) is used in this study [36, 37]. The KF uses the available observations and the predictions of previous tracks. It then obtains a maximum likelihood guess of the current state, taking into account the possibility of detection errors during the process. If no observation is detected, the KF estimate is used as the observed value. In SORT, the observed value is defined as the value of the bounding box obtained from the object detector. In this study, we focus on the position of a player projected onto the pitch plane as the observed value to be input to the KF. In this way, as shown in Fig. 6, even when no observed value is detected in the pitch plane, the predicted value of the KF is used to generate the movement trajectory.

When assigning detections to exist targets, the player’s state for each target is estimated by predicting the new position in the current frame. The assignment cost matrix is computed as the sum of the Euclidean distances between the predicted and detected values by KF. In this study, the linear assignment problem (LAP) to minimize this cost is solved optimally using the Hungarian method [38]. In this way, the prediction by the Kalman filter and the ID assignment of the object for each frame are made fast.

3.7 Players’ Pass Detection

As shown in Fig. 7, we detect the frames where the ball velocity peaks. Then, we assign team IDs and player IDs obtained in the previous subtasks to the detected frames. Then, we identify the passer and the receiver based on these IDs. We define the player who touches the ball as the player who is closest to the ball at the peak of the ball velocity. In this way, the passer is identified in the frame of the maximum value, and the receiver is identified in the frame of the minimum value.

3.8 Offside Algorithm

Using the outputs obtained from all the previous modules, we have information on the tracking positions of all players and balls, teams, passers, and receivers. This section transforms the offside rule, which also considers the time-series information, into a computational function for an accurate offside decision.

First, the final defender is identified on the pitch plane as the offside criterion. Then, in the frame, when the passer makes the pass, the offside candidate is an attacking player located in an area closer to the attacking team’s goal than the final defender. Finally, if the offside candidate receives the ball, the offside candidate is identified as offside.

4 EXPERIMENTS

4.1 Datasets

We have created and labeled our dataset because there is no publicly available dataset of multiple football matches recorded by fixed cameras. The dataset consists of match footage recorded using two fixed, high-resolution 30fps camera systems, positioned approximately 10m from the sideline in the middle of the football court. Two 4K AXIS P1447-LE cameras were used to film three 90-minute matches.

In order to evaluate the performance of offside detection, we cut out the same number of offside scenes and scenes without offside. In this paper, we manually cut out ten scenes of 30-second matches, including offside scenes and ten scenes without offside scenes from the three matches. The offside scenes were adjusted so that they were included in the 15-second frames. This is because it is necessary to consider the information of the time before the offside scene when making inferences. In addition, since the offside rule does not apply to corner kicks and throw-ins, the videos containing these cases were removed from the dataset.

4.2 Metrics for Evaluation

This subsection tests our proposed method on the dataset we have created and analyzes the results. We then discuss appropriate metrics to evaluate the offside detection task. In this study, we test our proposed method on ten different videos in the dataset, including scenes judged to be offside and ten different videos without offside scenes. We also discuss the improvements of the proposed method based on the obtained performances.

On the offside detection, To evaluate the performance of the proposed system accurately, we discuss the accuracy of the offside detection. Here, we need to take into account the distortion of the data. In each frame of each video, there may be only a few frames in

Algorithm 1 Offside Detection Algorithm

Input: Players, Passed Player, Received Player, OffenceTeamID, DefenceTeamID

Output: Offside Candidates, Offside Decision

```

// Find the position of 'Second-Last Defending Player'
1: DefencePlayerPos = []
2: for Player in Players do
3:   if Player[id] == DefenceTeamID then
4:     DefencePlayerPos.append(Player[Pos])
5:   end if
6: end for
7: Sort(DefencePlayerPos, Ascending)
8: OffsidelinePos = DefencePlayerPos[1]
//select the position of 'Second-Last Defending Player'
// At first check attacking team players in offside position when
// players passed
9: OffsideCandidate = []
10: for Player in Players do
11:   if Player[id]==OffenceTeamID then
12:     if Passed Player[id]==OffenceTeamID then
13:       if Player[Pos] < OffsidelinePos then
14:         OffsideCandidate.append(Player)
15:       end if
16:     end if
17:   end if
18: end for
// Check attacking team players for offside
19: for OffsideCandidate in OffsideCandidates do
20:   if OffsideCandidate == Received Player then
21:     OffsideCandidate[OffsideDecision] = ON
22:   else
23:     OffsideCandidate[OffsideDecision] = OFF
24:   end if
25: end for

```

the whole video in which an attacking player is placed in the offside position. Therefore, if the presence or absence of a player in the offside position is assessed for each frame, a very high TRUE FALSE value may be recorded, which may mislead the analysis results. Given this, in this study, we label the offside judgment results for a single match video scene and compare the labeled values with the judgment results of the detector. Table. 1 shows the comparison between the output result of the detector and the labeled result. Based on these figures, we consider the proposed system's accuracy, recall, and F1 score.

$$\text{Precision} = \frac{\text{Correct Offside Decisions}}{\text{Total Offside Decisions}}. \quad (4.1)$$

$$\text{Recall} = \frac{\text{Correct Offside Decisions}}{\text{Total Correct Offside Decisions}}. \quad (4.2)$$

$$F1 = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}. \quad (4.3)$$

5 RESULTS AND DISCUSSION

As a result, the proposed system records 60% for *Precision*, 66.7% for *Recall* and 63.1% for *F1Score*.

Table 1: Comparison of measured and predicted values for offside detection

	Positive(Real)	Negative(Real)
Positive(Pred)	$TP = 6$	$FP = 3$
Negative(Pred)	$FN = 4$	$TN = 7$

The following issues were qualitatively identified from the images of the offside scenes where the detector caused false positives. (1) the effect of noise when projecting from the image plane to the pitch plane and (2) the tracking accuracy in the path sequence.

The projection noise in the pitch-plane projection is caused by the inaccuracy of the annotations used to identify the keypoints. This inaccuracy can be attributed to the manual annotation of regions in the image far from the camera distance. As a result, the accuracy of the projection to the pitch coordinates decreases as the distance from the camera in real space increases. In this task, it is expected that the installation of fixed cameras on the opposite sides of the pitch will improve the accuracy of the whole system. In addition, in order to determine offside during a match, it is necessary to take into account that all body parts other than hands and arms are beyond the offside line [1]. Therefore, in future tasks, we will use posture estimation to obtain the position of each player's body part, which will allow us to detect offside more accurately and follow the IFAB rules more strictly.

As for tracking, we observed some false detections due to the wrong ID assignment of players. For example, in a pass sequence, the ID of an attacking player in the offside position was switched, resulting in a false offside detection. The misjudgments were caused by the tracking being performed only on the frames in the pass sequence that we had acquired in advance. The KF used for tracking determines the prediction of the next frame by using previous detections as observations [36]. Therefore, when the number of observations used is small, the predicted assignments may not be optimal. Thus, influencing the result of detection failure or false positive. This may be improved by adding a backward step to KF. It is possible to improve the accuracy of KF prediction of player positions by using other information other than just xy-coordinates on the pitch plane. For example, we consider that it is possible to prevent ID switching among teams by considering the team IDs obtained in the subtask of the team classification. Moreover, the use of image features obtained from pretrained deep convolutional neural networks may also be plausible.

6 CONCLUSION

This study proposes a method to automatically determine offside based on pass events by analyzing the match situation along the Spatio-temporal direction from football video. In this paper, we propose a system automatically judging offside based on the precise definition of offside. The F1 score obtained by this system is 63.1%. We have also identified the causes of the misjudgments of offside and have proposed solutions to them. This study also proposes a method to accurately track the ball's position about the players of both teams to detect offside accurately.

ACKNOWLEDGMENTS

JSPS KAKENHI Grant Number 19K22857 partially supports this research.

REFERENCES

- [1] IFAB, "Law11 offside," <https://www.theifab.com/laws/latest/offside/#offside-position>, 2021, [Online; accessed 2021].
- [2] P. Catteeuw, B. Gilis, A. Jaspers, J. Wagemans, and W. Helsen, "Training of perceptual-cognitive skills in offside decision making," *Journal of Sport and Exercise Psychology*, vol. 32, no. 6, 2010.
- [3] W. Helsen, B. Gilis, and M. Weston, "Errors in judging "offside" in association football: Test of the optical error versus the perceptual flash-lag hypothesis," *Journal of sports sciences*, vol. 36, no. 1, pp. 3–21, 2006.
- [4] B. Gilis, W. Helsen, P. Catteeuw, E. Van Roie, and J. Wagemans, "Interpretation and application of the offside law by expert assistant referees: Perception of spatial positions in complex dynamic events on and off the field," *Journal of sports sciences*, vol. 27, no. 1, pp. 551–63, 2009.
- [5] R. Oudejans, F. Bakker, R. Verheijen, J. Gerrits, M. Steinbrückner, and P. Beek, "How position and motion of expert assistant referees in soccer relate to the quality of their offside judgements during actual match," *International Journal of Sport Psychology*, vol. 36, pp. 3–21, 2005.
- [6] FootballTechnology, "VAR at the 2018 FIFA World Cup," <https://football-technology.fifa.com/en/innovations/var-at-the-world-cup/>, 2018, [Online; accessed 2021].
- [7] I. Erkekgorri, J. Castellano, I. Etxeazarra, and C. Peñas, "The effects of the video assistant referee system (var) on the playing time, technical-tactical and physical performance in elite soccer," *International Journal of Performance Analysis in Sport*, vol. 20, pp. 1–10, 2020.
- [8] elegraph Sport, "var in the premier league explained: How does it work, what decisions can be changed and why is it controversial?" www.telegraph.co.uk, 2019, [Online; accessed 2021].
- [9] C. Nee, "Is var causing more problems than it solves?" thesetpieces.com, 2019, [Online; accessed 2021].
- [10] N. Panse and A. Mahabaleshwarkar, "A dataset methodology for computer vision based offside detection in soccer," in *Proceedings of the 3rd International Workshop on Multimedia Content Analysis in Sports*, 2020.
- [11] S. Siratanita, K. Chamnongthai, and M. Muneyasu, "Saliency-based football offside detection," in *2017 17th International Symposium on Communications and Information Technologies (ISCIT)*, 2017.
- [12] S. Siratanita, "A method of saliency-based football-offside detection using six cameras," in *2017 Global Wireless Summit (GWS)*, 2017.
- [13] S. Siratanita, K. Chamnongthai, and M. Muneyasu, "A method of football-offside detection using multiple cameras for an automatic linesman assistance system," *Wireless Personal Communications*, vol. 118, 2021.
- [14] E. Lopez and P. Jenkins, "Offside detection system using an infrared camera tracking system," *World Journal of Mechanics*, vol. 09, pp. 163–176, 2019.
- [15] K. Muthuraman, P. Joshi, and S. Raman, "Vision based dynamic offside line marker for soccer games," 2018.
- [16] J. Singh, "Vision based method for offside detection in soccer." github.com/jagjeet-singh/Vision-based-method-for-offside-detection-in-soccer, 2017, [Online; accessed 2021].
- [17] B. Gilis, W. Helsen, P. Catteeuw, and J. Wagemans, "Offside decisions by expert assistant referees in association football: Perception and recall of spatial positions in complex dynamic events," *Journal of experimental psychology. Applied*, vol. 14, pp. 21–35, 2008.
- [18] T. D'Orazio, M. Leo, P. Spagnolo, P. L. Mazzeo, N. Mosca, M. Nitti, and A. Disante, "An investigation into the feasibility of real-time soccer offside detection from a multiple camera system," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 19, pp. 1804–1818, 2010.
- [19] P. Khirwadkar, "Offside detection system for football," github.com/kparth98/TTSP-Project, 2017, [Online; accessed 2021].
- [20] A. Deliege, A. Cioppa, S. Giancola, M. J. Seikavandi, J. V. Dueholm, K. Nasrollahi, B. Ghanem, T. B. Moeslund, and M. Van Droogenbroeck, "Socccernet-v2: A dataset and benchmarks for holistic understanding of broadcast soccer videos," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, 2021.
- [21] S. Giancola and B. Ghanem, "Temporally-aware feature pooling for action spotting in soccer broadcasts," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, 2021.
- [22] A. Cioppa, A. Deliege, F. Magera, S. Giancola, O. Barnich, B. Ghanem, and M. Van Droogenbroeck, "Camera calibration and player localization in socccernet-v2 and investigation of their representations for action spotting," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, 2021.
- [23] A. Bochkovskiy, C.-Y. Wang, and H. Liao, "Yolov4: Optimal speed and accuracy of object detection," in *ArXiv*, 2020.
- [24] Z. Dai, B. Cai, Y. Lin, and J. Chen, "Up-detr: Unsupervised pre-training for object detection with transformers," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.
- [25] J. Xu and K. Tasaka, "Keep your eye on the ball: Detection of kicking motions in multi-view 4k soccer videos," *ITE Transactions on Media Technology and Applications*, vol. 8, no. 2, pp. 81–88, 2020.
- [26] Z. Martin, S. Hendricks, and A. Patel, "Automated tackle injury risk assessment in contact-based sports - a rugby union example," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, 2021.
- [27] M. Koshkina, H. Pidaparthi, and J. H. Elder, "Contrastive learning for sports video: Unsupervised player classification," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, 2021.
- [28] R. Theagarajan, F. Pala, X. Zhang, and B. Bhanu, "Soccer: Who has the ball? generating visual analytics and player statistics," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPR)*, 2018.
- [29] D. Sorano, F. Carrara, P. Cintia, F. Falchi, and L. Pappalardo, "Automatic pass annotation from soccer videostreams based on object detection and lstm," 2020.
- [30] Z. Zhang, "A flexible new technique for camera calibration," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 22, pp. 1330–1334, 2000.
- [31] G. Jocher, A. Stoken, J. Borovec, NanoCode012, ChristopherSTAN, L. Changyu, Laughing, tkianai, A. Hogan, lorenzomammanna, yxNONG, AlexWang1900, L. Diaconu, Marc, wanghaoyang0106, ml5ah, Doug, F. Ingham, Frederik, Guilhen, Hatovix, J. Poznanski, J. Fang, L. Yu, changyu98, M. Wang, N. Gupta, O. Akhtar, PetrDvoracek, and P. Rai, "ultralytics/yolov5: v3.1 - Bug Fixes and Performance Improvements," 2020. [Online]. Available: <https://doi.org/10.5281/zenodo.4154370>
- [32] X. Yu, H. W. Leong, J.-H. Lim, Q. Tian, and Z. Jiang, "Team possession analysis for broadcast soccer video based on ball trajectory," in *Fourth International Conference on Information, Communications and Signal Processing*, vol. 3, 2003.
- [33] L. Dhakar, "color-thief," <https://github.com/lokesh/color-thief>, 2020, [Online; accessed 2021].
- [34] R. Nauss, "Solving the generalized assignment problem: An optimizing and heuristic approach," *INFORMS Journal on Computing*, vol. 15, pp. 249–266, 2003.
- [35] A. Bewley, Z. Ge, L. Ott, F. Ramos, and B. Upcroft, "Simple online and realtime tracking," *2016 IEEE International Conference on Image Processing (ICIP)*. [Online]. Available: <http://dx.doi.org/10.1109/ICIP.2016.7533003>
- [36] R. E. Kalman, "A new approach to linear filtering and prediction problems," *Journal of Basic Engineering*, vol. 82, no. 1, pp. 35–45, 1960.
- [37] W. Wang, N. He, K. Yao, and J. Tong, "Improved kalman filter and its application in initial alignment," *Optik*, vol. 226, pp. 165 747–165 751, 2020.
- [38] H. W. Kuhn and B. Yaw, "The hungarian method for the assignment problem," *Naval Res. Logist. Quart.*, pp. 83–97, 1955.