A new ball detection strategy for enhancing the performance of ball bees based on fuzzy inference engine

Arwa E. Abulwafa | Ahmed I. Saleh | Mohamed S. Saraya | Hesham A. Ali

Department of Computer Engineering and Systems, Faculty of Engineering, Mansoura University, Mansoura, Egypt

Abstract
Sports video analysis has received much attention as it turned to be a hot research area in the field of image processing. This motivation offers opportunities that develop fascinating applications supported by analysis of different sports, especially soccer. Ball identification, in soccer images, is an essential task not only for goal-scoring but also for performance evaluation. However, ball detection suffers from several hurdles such as occlusions, fast-moving objects, shadows, poor lighting, color contrast, and other static background objects. Although several ball detection techniques have been introduced such as Frame Difference, Mixture of Gaussian (MoG), Optical Flow, and so forth; ball detection in soccer games is still an open research area. In this paper, a new Fuzzy Based Ball Detection (FB^2D) strategy is proposed for identifying the ball through a set of image sequences extracted from a soccer match video. FB^2D can accurately identify the ball even if it is attached to the white lines drawn on the playground or partially occluded behind players. FB^2D is compared to recent ball detection techniques. Experimental results show that FB^2D outperforms recent detection techniques as it introduces both the highest level of detection accuracy in the testing stage and the lowest possible error.
1 | INTRODUCTION

Detection and tracking of the moving objects are principal tasks for Artificial Intelligence (AI). Accordingly, there must be an existence of a precise technique for the detection of moving objects to succeed in this mission. On the other hand, sport is equivalent to motion in which people and objects move at different speeds and directions. In the case of ball-based sports such as soccer,1 volleyball,2 basketball,3 tennis,4 and handball,5 the ball is always the main center of attention. If the ball is accurately detected even at high speeds, detection for the significant events of the game, metrics, and so much more beneficial information will be easy. Recently, there are several available techniques for detecting human bodies6 as detecting players is easy but not so the ball. There is an urgent need to suggest robust techniques for detecting the moving ball. A good ball detector must be efficient and fast. However, being able to detect the ball only is not sufficient. Late detection of the ball will not be helpful as it is impossible to make correct decisions or take actions based on this detection, because such action or decision will be useless.

The automatic analysis of soccer ball videos7 has become a significant point of the research endeavors as it is considered one of the most common group games around the world. There are many applications of the soccer ball (e.g., indexing, retrieval, annotation, summarization, event detection, and tactics analysis). Nowadays, playfield and ball detection always had much attention since a playfield is a place where almost all the events of the game take place, and the ball is the focus of both audience and players.

Throughout the soccer match, same as a ball game, players are continuously interacting with the ball. The ball remains the principal attention during the time of the game as most actions take place around it. Therefore, providing accurate ball detection, which is essential for ball tracking, is necessary for a comprehensive and deeper analysis of soccer games. Despite the higher quality hardware power and imaging devices, it is still difficult to accurately detect the ball within the frame of interest. For accurate ball detection, two issues must be considered; they are as follows: (i) a precise ball detection mechanism should be employed. (ii) it is important not only to get high-quality images but also to have images with explicit ball appearance as much as possible (i.e., the ball must not be occluded behind players). Concerning the first issue, different soft computing techniques such as fuzzy8 and neural networks9 could be employed to promote the performance of the detection system. On the other hand, resolving the second issue requires interactive photographing10 in which the camera is a movable one allowing continuous ball tracking. In interactive photographing, the object of interest (e.g., the ball) is under focus all the time. Accordingly, there is an excessive need for a camera, which continuously tracks the ball; it will be accomplished via ball bees.

This paper introduces a new Fuzzy Based Ball Detection (FB²D) strategy for detection of the soccer ball via Ball Bees. FB²D consists of two main phases: (i) Background Identification, and (ii) Object Detection. The main target of the first phase, carried out offline before starting the match, is to detect the background model. Although the dominant color of the background is green, it is needed to identify the exact greening range of the playground. The intensity and degree of the playgrounds’ greening vary according to the color of the grass, which represents the playing field as well as the intensity of lighting. The color of the grass may be different even in the neighboring
parts of the stadium. On the other hand, the intensity of lighting varies in sunny and shady places, whether the match takes place during the day or at night as well as the distance from the lighting towers. Initially, to identify the greening range of the playground, some frames of the empty playfield are taken using a background Scanner Bee. The Scanner Bee moves diagonally from the corner to the playground center points taking a set of consecutive frames. And then, the white pixels included within these frames will be excluded. Consequently, these frames are converted from RGB color space to HSV, and the hue component will be arranged in a vector called H_Vector. These hue values will be filtered by removing both duplications and outliers. Ultimately, a range of the hue component of the playfield greening color is formed.

On the other hand, the main target of the object detection phase, activated as long as the match continues, is to detect the ball in the current frame. It consists of four sub-phases: Background Subtraction, Edge detection, Ball Candidates Identification, and Classification. During the first subphase, the RGB color space will be converted to HSV one, then its hue component is considered. This hue component is compared with the predefined hue values of the greening range stored in the H_Vector. Moreover, the matching values will be removed from the current frame to exclude the playfield background color. During the edge detection subphase, the Canny edge detector is applied to detect the frame objects. It works by detecting discontinuities in brightness and finding the boundaries of objects within the image. During the third subphase, a set of candidate balls are identified from the objects found in the frame using the Circle Hough Transform technique. Subsequently, a new-fuzzy based inference engine, during the last subphase, is used to confirm the actual ball. Three different fuzzy sets are considered; they are Form Factor FF, Area Ratio AR, and Eccentricity EC. Employing fuzzy logic has a significant impact on promoting performance as it maximizes ball identification.

The proposed FB^2D has been compared to contemporary soccer ball detection strategies; the experimental results demonstrate that FB^2D could be used to support object detection and tracking as it outperforms recent detection strategies in terms of precision, recall, and detection accuracy. The rest divisions of this paper are as follows: Section 2 introduces the background and basic concepts, Section 3 presents the previous efforts in the area of ball detection, Section 4 discusses the problem statement, Section 5 shows the proposed system, Section 6 presents the performance analysis and experimental results, Section 7 introduces the pros and cons of our proposed system, and Section 8 summarizes our conclusions.

2 | BACKGROUND AND BASIC CONCEPTS

In this section, a brief explanation of the camera setup for ball detection and tracking will be introduced. Also, the general procedure for soccer ball detection will be introduced. Also, an explanation of the ball bee will be presented here. A ball bee is a distinctive type of quadcopter^11; it can track the moving ball. It was firstly introduced by Abulwafa et al.12 and used to track the moving ball in soccer matches. A ball bee is a quadcopter with an integrated moving camera situated directly below it. Its camera can take high-quality photos from altitude and record clear shots. The camera frame is made of lightweight, composite materials to reduce weight and increase flight maneuverability. The operator uses a remote control to launch, navigate and land remotely. Controllers use radio waves to communicate with the bee via Wi-Fi. It has many working sensors onboard (e.g., speed and distance sensors, infrared and thermal sensors, image sensors, chemical sensors, GPS, etc.).13,14 There exists an engagement between the used ball detection technique and the hosting ball bee in a mutually beneficial
relationship. Confidently, the employed ball bee constitutes a source of high-quality images in which the ball is not occluded as possible, which simplifies the task of the driving ball detection technique. On the other hand, the ball detection technique is used to derive the ball bee itself. Hence, employing an accurate ball detection technique will promote the performance of the ball bee, which in turn increases the accuracy of the driving detection technique.

Ball detection is an essential step of action recognition and ball tracking. Simply, it is necessary to detect the ball position to track it. The localization of the ball is not only the main factor in tracking performance but also in detecting the main events of the game. Despite it looks simple, discovering the soccer ball in video frames is not a trivial task for the following reasons: (i) the ball, normally, has a spherical shape, but it might be observed as a blurred object due to its rapid motions and unpredictable acceleration. For the same reason, the static features of the ball might not be correctly observed, (ii) the ball state changes during the game as it is thrown, hit, or kicked by players, (iii) the ball is the smallest element on the game field; hence, it might be confused with other objects on the playing field such as penalty shooting point, field lines, or players parts etc., (vi) during the game, the ball is usually occluded by players as it disappears behind the feet of the players most of the game time, (v) sometimes when the ball is kicked high in the air, it suddenly gets out the camera field of view, (vi) it is difficult to detect the playfield due to the shadow from illumination sunlight, (vii) the features of the ball such as color, shape, and size are changing ones with the situation conditions like light and velocity, (viii) the difficulty of segmentation the ball from the player when the ball is possessed by players, and (ix) the fragments because of inappropriate segmentation of players or lines of the field may resemble the ball. Recently, many techniques for soccer ball detection in video frames have been introduced in the literature. Some of those detection techniques include single view mono cameras while the others employ multiple and mostly dedicated cameras (e.g., offside cameras). The most successful techniques in the literature rely on two separate sequential phases, namely: (i) ball candidate extraction, and (ii) ball candidate validation. The purpose of the first phase is to elect the regions that most probably contain the ball, while during the second phase, the candidate regions are accurately analyzed to recognize which of them contains the ball.

2.1 Camera setup for ball detection and tracking

A pivotal part of any ball detection system is the setup of the camera as one with high image quality, and a clear field of view will make the detection process smooth. The position of the cameras must cover the complete playfield and never overlap with the game in any way. Frequently, other objects of interest such as the players and the court lines also have to be detected besides the ball, as well as the camera setup has to be considered.

As illustrated in Figure 1, four different approaches can be used for ball detection and tracking; they are as follows: (i) Single Camera Approach (SCA), (ii) Multiple Camera Approach (MCA), (iii) Broadcast TV Approach (BTVA), and (iv) Ball Bee Approach (B₂A). According to SCA, only a single camera is employed in the detection and tracking process due to its easy use. Polceanu et al. employed it, where a Raspberry Pi with a single wide-angle fish-eye lens camera is used to record the videos. Also, the work introduced by Yan et al. used a single-camera view from behind the tennis court. On the other hand, MCA is mostly used to deal effectively with the occlusion or for mapping larger regions. Even though MCA offers a considerable advantage in defining the ball position, its setup is more complex than SCA. Owens et al. used multiple cameras with high performance, fixed in the tennis court for
broadcasting purposes. Nine IP cameras and one overhead camera were utilized by Conaire et al. This setup covers the whole court area from all the court corners. The system introduced by Fazio et al. utilizes two dual-camera smartphones placed behind the court along the sidelines to record the videos. Using accurate cameras of smartphones is an appropriate approach as they give high-quality imagery for ball detection.

Another way for obtaining a data set for both ball detection and tracking is BTVA, which utilizes the broadcast TV videos of the sport. Although BTVA makes it easier for obtaining game videos, it suffers from various camera angles and pan, tilt, zoom, and motion of the camera in a broadcast video. The last approach that can be used for ball tracking is B2A, which utilizes a set of cooperative ball bees for tracking the ball across the football court. Abulwafa et al. introduced it, where a quadcopter, called the ball bee, has been used for tracking the soccer ball. The camera of this bee is a moving camera fixed in the bottom body of it; also, the photos taken by this camera are high-quality photos.

### 2.2 General procedure for soccer ball detection

As shown in Figure 2, ball detection can be carried out through three phases: (i) Pre-Processing Phase (PP), (ii) Ball Candidate Phase (BCP), and (iii) Ball Detection Phase (BDP). During PP, two essential tasks should be done for system setup; they are background and ball modeling. Concerning the background modeling, a set of input frames (initial frames) are employed to model the dominant background, while the employed ball (maybe a set of balls) is also considered to build the ball model, which will be employed through the following phases to identify the candidate balls. The generated ball model summarizes the ball features such as size, color, shape, and so forth. After system setup (e.g., PP termination), the system will be ready to start. BCP and BDP are the two sequential phases followed to detect the ball in each frame.
The main target of BCP is to generate the ball candidates. During BCP, the background is subtracted from the input frame to detect the frame foreground. An appropriate segmentation technique is then applied to detect foreground shapes. After segmentation, morphological operations should be employed to bridge the gaps as well as eliminate the noise. To identify the movable objects, the background, an edge detection technique is exploited. The result of segmentation is a group of ball candidates and other objects that had looked like balls. Then, circularity measures are applied to identify candidate balls. These ball candidates should be tested in terms of the ball features extracted during the PP phase through the ball modeling process to identify the correct ball if exists. This can be carried out during BDP through the ball validation process. Finally, the ball can be localized within the frame to update the template and to guide later possible ball re-detection.

3 | RELATED WORK

In soccer videos, the most significant object is the ball where various applications stimulate ball detection; these applications are event detection, tactics analysis, automatic summarization, and object-based compression. Fu and Han update the background pixels using a recursive filter that only depends on the learning parameters, which may delay the background model updating. Yao and Ling present an improvement of GMM, however, it cannot detect the object near the camouflage area. The fast-learning rate techniques may delay the object detection and make the accuracy low. On the other hand, techniques
depending on the low learning rate do not upgrade the background model effectively. Thus, it may be valuable to use the regional level of processing to update the changing background pixels and to find the exact moving ones. Kumar and Yadav\textsuperscript{28} detect the true nonstationary pixels from the foreground pixels using the regional level of processing by implementing the entropy and the energy.

Many papers have presented different methods to form background models, as a perfect background subtraction method makes the results of object detection better. For example, Vo and Park\textsuperscript{29} introduce a method to determine the variation of the background intensity for an image and eliminates this determination from the selected image. As the computational cost of this technique is high, it is not appropriate for real-time applications. Cheng et al.\textsuperscript{30} present an image pyramid form consists of two layers to find the sum of absolute differences (SAD) value for images with low dimensions. Also, the color variations between the foreground and background are modified to distinguish between nonstationary objects and noise. Additionally, this technique cannot design the background if it is a changing one. Clustering is another way to create a background model.

A ball detection procedure for actual images under the conditions of changeable light and unstable backgrounds has been proposed by D'Orazio et al.\textsuperscript{31} Both computational costs and the performance of the system have been considered in the applied technique. While Moyyila\textsuperscript{32} introduces a SURF algorithm, it has fast and robust features to detect the ball and display its information. This algorithm shows the ability and robustness of detecting objects. Chiu et al.\textsuperscript{33} propose a background subtraction technique called Entropy-Based Initial Background Extraction (EBBE), used for limited background conditions. Also, the production of false foregrounds has been decreased by this technique in situations of highly color interferences either due to complex conditions such as camera shake and shaking trees or due to a very low frame rate. A unified model technique called Yolov3—Improved Non-Maximum Suppression (INMS) is introduced by Bondalapati et al.\textsuperscript{34} to improve the performance of the detection of the moving objects. Balaji et al.\textsuperscript{35} propose a Metaheuristic algorithm, called the Cuckoo Search Algorithm, for the detection of the objects from the sports videos. This algorithm also deals with object detection challenges such as shadows and lighting; it achieves high accuracy and precision results.

An automatic background subtraction technique is presented by Huang et al.,\textsuperscript{36} where a histogram learning technique is used to detect the background pixels. The foreground and background pixels are determined using color models via a training set of soccer videos. Another algorithm is proposed by Ali et al.,\textsuperscript{37} where the background color is supposed to be a green color. Mudjirahardjo et al.\textsuperscript{38} and Mudjirahardjo et al.\textsuperscript{39} introduce a method in which the foreground is subtracted from the background by using the Euclidean distance function of the selected pixels. To decide whether the selected pixel is a foreground or not, the distance must be less than a specific threshold. Mudjirahardjo et al.\textsuperscript{38} calculated the distance in RGB color space. This distance, on the other hand, is calculated in HSV color space by Mudjirahardjo et al.\textsuperscript{39} Chakraborty and Meher\textsuperscript{40} propose a background subtraction strategy and a frame difference strategy over three sequential frames to detect moving objects. The background extraction is carried out by Durus\textsuperscript{41} using the dominant color distribution. The target object has been determined by Mudjirahardjo et al.\textsuperscript{42} using the velocity histogram.

A deep learning strategy for 2D ball detection and tracking (DLBT) in soccer videos is reported by Kamble et al.\textsuperscript{43}; it faces several challenges. For blob recognition of moving objects, a new 2-stage buffer median filtering background model is applied. Initially, a deep learning
strategy is proposed for classifying an image patch into three classes: ball, player, and background. To identify the ball from the earliest frames, DLBT does not require human intervention. A deep network-based object detector specialized for ball detection in long shot films is described by Komorowski et al. The algorithm works on images of any size and creates a ball confidence map encoding the position of the identified ball due to its completely convolutional nature. Hiemann et al. present a real-time ball detection strategy based on the YOLOv3 object detection model that addresses the detection of small and fast-moving balls in sport video data. To improve detection accuracy and speed, they make critical changes to the network design and training procedure.

As depicted in Table 1, the traditional ball detection strategies perfectly deal with circumstances where the ball is visible as a single object and separated from the player body. On the other hand, these strategies have issues for detecting the ball when it is possessed or partially occluded by a player.

4 | PROBLEM STATEMENT

Generally, the goal of motion detection is the determination of the moving points or regions in the image that have moved between the two-time instants. The motion of these points or regions is not directly recognized but rather through intensity changes. The changing of intensity through time may also be happened by camera noise or illumination changes. Moreover, the motion of the object may make minor intensity variations or even none at all. The automatic moving objects' detection is the first critical phase of different and distinguish applications such as pedestrian and vehicle tracking, action and event recognition, video annotation. Undoubtedly, moving objects' detection is a vital topic in the computer vision field. For instance, applications like intelligent surveillance systems, ball recognition and tracking, human-computer interface (HCI), and robot visions need moving objects' detection process. Many different strategies have been presented for the detection of moving objects in the case of stationary cameras. On the other hand, these strategies do not perform effectively when mobile or pan-tilt-zoom (PTZ) cameras are used, because of the unconsecrated factors of movable cameras.

The essential capability of any system is the ability of both accurate ball detection and tracking in a video sequence; it aims to an automatic analysis of football matches or players' progress. Unfortunately, ball detection through long-shot video frames of a soccer game is a significant task for the following reasons: (i) the object of interest (e.g., the ball) has a very small size compared to other visible objects in the observed scene. (ii) The size of the ball differs according to its location on the court due to the perspective projection. (iii) The ball is not perfectly in a circular shape. (iv) While kicking the ball, it moves with high velocity; consequently, it turns into a blur and elliptical shape. (v) The visual color of the ball varies according to the changing of shadows and lighting. (vi) The ball color is similar to the color of white field lines and sometimes to the jerseys of the players. (vii) Other objects can be similar to the ball, as small areas near the field lines and parts of players' bodies such as their heads. (viii) The ball can be partially occluded by players' bodies. Figure 3 illustrates a high variance in the ball appearance and accordingly the difficulty of the ball detection task.
Ball detection can be defined as the discovering of circular characteristics in images’ sequences when the ball seems like a circular shape, which can be achieved by light conditions. In the case of using artificial lights during the evening matches or the shadows of the sunlight during the noon matches, the ball shape is more spherical. The ball shape and its direction differ based on the direction of the light in the playfield. Thus, the ball detection and recognition methods must be suitable to detect the different shapes of the ball in the image. There are many different methods for moving object detection: (i) optical flow technique, (ii) consecutive frame difference, and (iii) optical flow technique.

| Year | Paper | Contribution | Advantages | Disadvantages |
|------|-------|--------------|------------|---------------|
| 2021 | [45]  | A real-time ball detection strategy based on the YOLOv3 object detection model. | Improve detection accuracy and speed | A multi-stage training process is difficult to handle |
| 2020 | [46]  | Proposed a Metaheuristic algorithm for the detection of the sports objects from the video sequence. | • Rectified the challenges in object detection due to shadows, lighting, and • Detected the fast-moving player while hitting the ball, etc. | • Occlusion problem • Efficiency Problem |
| 2018 | [47]  | Estimated the background intensity variation of an input image and Subtracted the estimate from the input to achieve a flat-background image. | Has great values in robust Image binarization and image segmentation. | • High computational cost • Not suitable for real-time systems. |
| 2017 | [45]  | Integrated the regional level processing by evaluating the entropy and energy that provided the actual moving pixels on the foreground. | Efficiently localizes the object in the scene. | Not suitable for multiple object detection and tracking in unconstrained videos. |
| 2015 | [48]  | SURF algorithm is used for each frame image, to find object quickly and display its information. | SURF is preferred over SIFT | Cannot detect and differentiate player according to team. |
| 2014 | [44]  | Constructed an improved version of GMM with frame difference method to the foreground extracted. | • Used for very crowded situations. • Proposed a new shadow removal method based on RGB color space is. | Fails to detect the object near camouflage region. |
| 2012 | [37]  | Detected the soccer ball and players within a video footage. | Used especially when the ball is attached with lines in the ground | Not suitable for real-time applications. |
Automatic extraction of the object features has been performed using a clustering method in the Optical Flow technique. Thus, the features extraction process is done using an x-mean cluster, and then these extracted features will be classified based on their predefined motion parameters and labeled as moving objects. Besides, this technique needs some additional components to be employed in real-time systems.

The second technique, called Consecutive Frame Difference, is a simple technique suitable in the dynamic environment, even though it cannot detect the moving object perfectly. The background subtraction technique is the most popular one in which input frames are compared with the background model to extract the moving object. Although this method is accurate and has low computational time, its implementation in changing environments is more difficult due to the illuminations and lights variations, shadows, colors similarity, and so forth. The proposed strategy in this paper is called FB2D strategy. It consists of two main steps, as shown in Figure 4; they are Background Identification and Object detection. These two steps will be explained in the following sections as shown in Algorithm 1.

### 5.1 Background identification

The process of identifying the background is an essential step for detecting the objects. It has been applied in many studies and papers such as Monnet et al.\(^47\) directly addressed the modeling of dynamic background by proposing an auto-regressive model to predict the behavior of such backgrounds for effective detection of moving objects. For modeling and classification of dynamic background, the dynamic texture modeling methods can also be utilized. Parameters of such a background model can be effectively used as a cue for moving object detection. Recently, Minematsu et al.\(^49\) suggest an adaptive background model registration based on homography motion estimation to handle highly complex backgrounds for detecting moving objects for moving cameras. In general, the detection of moving objects in complex backgrounds requires good modeling of the background and precise estimation of camera motions.

In this paper, identifying the background model is essential due to the following: (i) simplifying the ball detection process in the case of a moving camera. (ii) taking into account the...
FIGURE 4  Fuzzy based ball detection (FB²D) strategy framework [Color figure can be viewed at wileyonlinelibrary.com]
robustness against illumination changes. In particular, soccer matches are played in outdoor playfields and the light intensity usually varies; therefore, a robust background modeling method must be able to adapt to gradual changes of the light in the environment. (iii) the playfield may be dynamic, namely some regions of the background may contain movement. An excellent background modeling method should effectively identify the periodical or irregular movement of objects. (iv) removing the shadows of foreground regions or ignore these irrelevant shadows. As the foreground objects often have shaded areas owing to the influence of light changing, which usually affect the separation of foreground objects and the performance of subsequent modules of a background modeling algorithm. (v) considering the effects of noise and being able to cope with degraded signals affected by different types of noise caused by camera shaking, lens ageing or sensor noise can cause image degradation.

Algorithm 1. Fuzzy based ball detection (FB^2D) strategy

| Algorithm Parameters |
|----------------------|
| Inputs: |
| • Frame Set of empty playfield FE (taken before the match starts) |
| • Frame Set F of the playfield (taken during the match) |
| • Range of ball radius [Rmin, Rmax] |
| Output: |
| The object is a Ball or not |
| Steps: |
| // Background Identification |
| 1. For each frame f0, FE Do |
| 2. f0=rgb2hsv (f0) // Convert to HSV |
| 3. f0=fgbg2hsv (f0) // Convert to HSV |
| 4. H(f0)=f0(:,1) // Identify Range H(f0) |
| 5. Hvector = Hvector + ∑I∈H(f0) // Add Range H(f0) to H_Vector |
| 6. End For |
| 7. Remove duplications from H_Vector |
| 8. Sort ascendingly of H_Vector |
| 9. H vector = H vector - out // Remove outliers from H Vector |
| // Object Detection |
| 10. For each frame f ∈ F Do |
| 11. f = rgb2hsv (f) // Convert to HSV |
| 12. f = f - H vector // Remove similarities |
| 13. f = hsv2rgb (f) // Convert to RGB |
| 14. f = rgb2gray (f) // Convert to Gray level |
| 15. f = edge(f, 'Canny') // Apply Canny Edge Detector |
| 16. BC= circle_hough (f, [Rmin, Rmax]) |
| 17. For each ball candidate bc ∈ BC Do |
| 18. Calculate membership values |
| 19. Calculate fuzzy output of each rule as; |
| 20. μ_j (x) = MAX (MIN (implication (μ_imp(j), μ_env(x)))) \forall x \in \{1,2,3,\ldots,n\} |
| 21. Calculate the output crisp value using the center of gravity method as; |
| 22. Cg = \sum (SV) * SV / μ(SV) |
| 23. Boll isball // define a flag called isball |
| 24. If COG ≥ β Then |
| 25. isball = true //bc is the ball |
| 26. Goto step 29 |
| 27. Else |
| 28. isball = false //bc is not a ball |
| 29. Goto step 15 |
| 30. End IF |
| 31. End For |
| 32. End For
Soccer matches are normally played on a grassy playfield; it is defined as the part of the image on which the match is taking place. To detect the ball, the first and useful step is to detect the playfield pixels. The soccer playfield has one distinct and dominant color, a tone of green color that may vary from a playground to another, in addition to the weather conditions and lighting variations within the same playground. Thus, any specific value for the field color does not be assumed in the suggested algorithm. This section delineates a method for defining the values of the playfield color. As the standard RGB color representation is not suitable, the RGB values of the captured images are converted to the corresponding coefficients in the HSV color space to avoid the influence of illumination before the analysis process. Furthermore, HSV represents three diverse parameters and they are as follows: hue, saturation, and brightness. The hue component is used to determine the dominant color wavelength, which has values from 0° to 360°. On the other hand, brightness illustrates the white light level ranging from 0 to 100, while saturation defines the chromatic element proportion in a color. The color depth of the soccer playfield is ranged from dark green to bright green.

In this step, it is assumed that the playfield has 4 lighting towers, as shown in Figure 5, and each tower has 16 lamps so the total number of lamps will be 64. Our proposed algorithm for playfield background detection consists of five steps: firstly, before the beginning of the soccer match when the playfield is empty, one of the bees, which will be called Scanner Bee, will fly and take a set of frames $FE \in \{fe_1, fe_n\}$ along the diagonal line between one corner and the center of the playfield as shown in Figure 6. This Scanner Bee is responsible for discovering the background model through the movement from the corner to the middle of the field, before the start of the match, and it can work again during the match if the lighting changes, especially when switching from natural lighting to artificial lighting. Secondly, the white pixels that may appear in the taken frames will be removed by using a threshold value. If the values of $R$ (red) or $G$ (green) or $B$ (blue) of any pixel in the frame exceed the threshold value as in (1), these pixels will be removed from the frame.

$P(x, y) = \begin{cases} 1 & \text{if } R(x, y) \geq \beta \text{ or } G(x, y) \geq \beta \text{ or } B(x, y) \geq \beta \\ 0 & \text{otherwise} \end{cases}$, \hspace{1cm} (1)

where $P$ is any pixel in the frame $fe_i$, $(x, y)$ is the pixel location—where the origin of the frame is in the top left corner—, $R(x, y)$, $G(x, y)$, and $B(x, y)$ are the red, green and blue component of $(x, y)$ pixel, respectively and $\beta$ is the threshold value. In (1), “1” refers that the pixel has white color and “0” refers that the pixel is not white pixel. Consequently, any pixel has $P$ equal to “1” will

FIGURE 5  The soccer playfield [Color figure can be viewed at wileyonlinelibrary.com]
be removed from the frame. On the other hand, any pixel has $P$ equal to “0” will be saved to be used in the next steps.

Thirdly, the filtered frames from the previous step will be converted from RGB to HSV color space. This conversion is important as the HSV color space is a device-independent color representation format. It has less noise than the RGB color images, and it is useful for detecting specific color types (e.g., skin color, fire color, grass color, etc.). Although different color variation can be seen in these areas, the hue for all these regions do not vary much, so hue values can be useful in soccer playfield identification. The formulae for converting RGB to HSV color space can be referred to [51]. Fourth, for each frame $f_e$ in the frame set $FE$, the value of the hue component for each pixel $P$ will be determined to identify the range of hues, which will be called $Range_H(f_e)$. This range will be added to the $H_Vector$, which is a vector containing all the values of the hue component in all frames of the frame set $FE$. And then, the duplicated values will be removed from this vector and they will be sorted ascendingly. Finally, the extreme values, at the fifth step, will be considered as outliers and they have to be removed from the $H_Vector$ using the Interquartile Range IQR technique. These outliers can come from anything seen in the playground. In Figure 7, the playfield looks like an empty field, but it has some unwanted objects, which will make some extreme values (outliers) in the $H_Vector$. These values are not green pixels, so they have to be removed from the $H_Vector$.

Any set of data can be described by its five-number summary. These five numbers, which give you the information you need to find patterns and outliers, ascendingly consist of: (1) the minimum or lowest value of the data set ($L$), (2) the first quartile $Q_1$, which represents a quarter of the way through the list of all data, (3) the median of the data set, which represents the midpoint of

![Figure 6](wileyonlinelibrary.com)  
**Figure 6** The frames along the diagonal from one corner to the center of the playfield [Color figure can be viewed at wileyonlinelibrary.com]

![Figure 7](wileyonlinelibrary.com)  
**Figure 7** Outliers in the soccer playfield [Color figure can be viewed at wileyonlinelibrary.com]
the whole list of data, (4) the third quartile $Q_3$, which represents three-quarters of the way through the list of all data, (5) the maximum or highest value of the data set ($H$). These five numbers tell a person more about their data than looking at the numbers all at once could or at least make this much easier. Interquartile range (IQR) is a technique that helps to find outliers in the data continually distributed. IQR is used to divide the whole data set to build a boxplot, a simple graphical representation of interquartile range. In this method, the data set is divided into quartiles and orders the data set into four equal parts. The divided ranges are $Q_1$, $Q_2$, and $Q_3$ named as first, second, and third quartiles respectively. Figure 8 shows how the boxplot is built. To find the IQR, the first quartile is subtracted from the third quartile ($IQR = Q_3 - Q_1$).

The IQR shows how the data is spread about the median. The interquartile range can be used to detect outliers, which is fulfilled using the following steps: (i) calculate the IQR for the data. (ii) calculate the highest value of the data set by multiplying the interquartile range (IQR) by 1.5 (a constant used to discern outliers), then add $1.5 \times IQR$ to the third quartile $Q_3$; consequently, any number greater than this is a suspected outlier. (iii) calculate the lowest value of the data set by subtracting $1.5 \times IQR$ from the first quartile $Q_1$, so any number less than this is a suspected outlier. By defining the normal data range with a lower limit as $L = Q_1 - 1.5 \times IQR$ and upper limit as $H = Q_3 + 1.5 \times IQR$, any data point outside this range is considered as an outlier and should be removed for further analysis. For instance, some samples for odd entries with 13 rows are used. The samples must be arranged in ascending order as follows: $-4, 4, 12, 14, 19, 22, 22, 24, 35, 44, 48, 83, 99$. The first quartile $Q_1 = 13$ and the third quartile $Q_3 = 46$ then the $IQR = Q_3 - Q_1 = 46 - 13 = 33$. Now, the lower boundary and upper boundary are calculated to define the outlier calculation. The value that is less and greater than these boundary values is considered as an outlier. Lower boundary $L = Q_1 - 1.5 \times IQR$ and $L = 13 - 1.5 \times 33 = -36.5$; upper boundary $H = Q_3 - 1.5 \times IQR$ and $H = 46 + 1.5 \times 33 = 95.5$.

![Figure 8](https://wileyonlinelibrary.com)
5.2  |  Object detection

In this section, a new ball detection algorithm will be introduced. The proposed technique consists of four main steps, and they are as follows: Background Subtraction, Edge Detection, Ball Candidates Identification, and Classification using a Ball Fuzzy Inference Engine. These steps will be explained in detail in the following sections.

5.2.1  |  Background subtraction

The background subtraction method is mainly employed to detect a moving object in an image by comparing two different frames. The difference value is compared with a specific threshold value calculated via the first few frames that are given. Object detection is used to check whether an object exists in a video or not as well as locating it. Moreover, the most popular challenge facing the background subtraction method is frequent changes due to illumination changes, motion changes, and background geometry changes. There are various background subtraction methods such as frame difference, Gaussian mixture model, kernel density estimation, codebook, and so forth.

In this step, the suggested background subtraction method consists of four steps. In the first step, when the soccer match starts, all the bees will fly to search for the ball in the playfield and take frames with their cameras. While in the second step, these taken frames will be converted into HSV color space. For each frame, the hue component of each pixel will be determined in the third step. And then, these values will be compared to the predefined H_Vector in the background model from the previous section. Finally, in the fourth step, the pixels with the same predefined values will be defined as the background pixels in the frame. Hence, these background pixels will be removed from the frame. As shown in Figure 10C, the remained pixels in the frame will be the moving objects, which are the players and the ball.

5.2.2  |  Edge detection

The Edge detection process can be defined as the determination of the sharp color discontinuities, which are the sudden variation in pixel intensity of an image. This process is used for both image segmentation and data extraction in fields like image processing, computer vision, and machine vision. Edge detection has many applications in the image processing field such as object recognition, motion analysis, pattern recognition, medical image processing, and so forth. There are two basic standards, and the first standard determines the areas that have a magnitude of the first derivative of the intensity greater than a certain threshold. The second one locates the areas where the second derivative of the intensity has a zero crossing. Generally, the edge function can be calculated, as in the following equation:

\[
[g, t] = \text{edge}(I, 'M', \text{params})
\]

Where “I” is the input image, ‘M’ is one of the edge detection techniques, and “params” are additional ones. In this paper, the Canny edge detector is employed to detect the soccer ball. Edges in this method can be determined by isolating noise from the image without
affecting the edges' features and then applying the tendency to define the critical threshold. In this section, the Canny edge detector will be used for defining the edges of the moving objects, as illustrated in Figure 9F.

The algorithm of the Canny edge detector has three phases: first, the noise of the image has been filtered out by using a Gaussian smoothing filter. Second, the edge intensity in the image is defined by finding the gradient of the image, which determines the place of the actual edge. Thirdly, the algorithm uses the non-maximum suppression method, which traces all pixels in the edge and set any pixel that is not at the maximum to be 0, to thins down the edges. Eventually, pixel connectivity and double threshold are used to determine and connect the edges. This means that a pixel can be considered as an edge if its magnitude is above the specified high threshold. On the other hand, a pixel can be defined as a non-edge if the magnitude is below the specified low threshold. The maximum number of edges can be detected via this edge detector, and it can also discover both the corner edges and the circular edges; consequently, it is suitable for soccer ball detection.

The Canny edge detector is the most well-defined and commonly utilized of the edge detectors. Canny proposed a set of edge detection goals and described the best way to achieve them. The first concern is good detection (low error rate), in which the edge detector should respond only to edges and should find all of them; no edges should be missed, according to Canny. The second problem is determining good spatial localization, which is defined as the distance between edge pixels. It also calculates the reciprocal of the root-mean-squared distance of the marked edge from the true edge's center as localization improves. The third concern is a high response rate; it means that the edge detector should detect several edge pixels even if only one is present. The first criterion assumes only one reaction to a single edge. However, it is made to explicitly reject multiple answers. The zero crossings of the image’s Laplacian of Gaussian (LOG) can be used to verify that closed edge contours are obtained. As a result, the Canny edge detector employs probability to determine error rate, localization, and responsiveness, as well as to improve the signal to noise ratio, detection, and noise sensitivity.
5.2.3 | Ball candidates identification

After finishing the Edge Detection step, only the objects that need to be considered and to be examined in the playfield will be the ball, the referee, the players, and some noises. Thus, the Circle Hough Transform (CHT) technique\(^5\) will be used to discover the circular objects with a predefined radius \(R\). The CHT technique is used to detect imperfect circles in images. Each circular edge with radius \(R\) will be a point in an accumulator matrix. The circle candidates are generated by casting “votes” in the Hough parameter space. Later, the element with the greater number of “votes” will be the center of the required circular object. To use the CHT technique, the range of the expected ball radius values needs to be identified as shown in Figure 10. Consequently, before the soccer ball match starts, the Scanner Bee will take two frames of the ball. In one of these two frames, the ball has to be in the center of the frame to determine the maximum radius \(R_{\text{max}}\) of the ball. On the other frame, the ball will be in the corner of the frame to find the minimum radius \(R_{\text{min}}\). From these two positions of the ball in the frame, the range of ball radius values will be determined as \((1 - \rho) \times R_{\text{min}} \leq R \leq (1 - \rho) \times R_{\text{max}}\), where \(0 \leq \rho \leq 1\).

5.2.4 | Ball Fuzzy Inference Engine (BFIE)

After ball candidates have been identified over the current frame, a fuzzy inference engine is used to decide whether the candidate is a ball or not. In this engine, the contour of each candidate is outlined and used to analyze and describe its shape features, which will be the Fuzzy Sets. In this paper, three features will be used to identify the ball candidates; they are Form Factor, Eccentricity and Area Ratio, which are explained through the following definitions.

**Definition 1.** Form Factor, denoted as FF (BC); measuring the degree of roundness of the ball candidate, and is defined as the ratio of 4 multiply \(\pi\) multiplies the area of the ball candidate to the square of the perimeter of that candidate.

**Definition 2.** Eccentricity, denoted as EC (BC); measuring the compactness of the ball candidate, and is defined as the distance from the center to the focus divided by the major axis of the candidate object.

**FIGURE 10** Identify the range of ball radius values [Color figure can be viewed at wileyonlinelibrary.com]
Definition 3. Area Ratio, denoted as AR (BC); measuring the ratio of ball candidate area to its minimal bounding rectangular (MBR) area.

The first feature is the Form Factor, which will be calculated as in the following equation:

\[ FF = 4\pi A/P^2, \] (3)

where \( P \) and \( A \) constitute both the perimeter and the area of the object, respectively. Commonly, the more circular the shape is, the closer to one the form factor will be. For instance, in case the object is a circle with radius \( r \) as in Figure 11A, and then its perimeter will equal to \( 2\pi r \); its area will be \( \pi r^2 \). By substituting in (3), the form factor (FF) will be

\[ FF = 4\pi r^2/(2\pi r)^2 = \frac{4\pi r^2}{4\pi r^2} = 1. \]

On the other hand, if the object is a square with length \( L \) as in Figure 11B, the form factor will be

\[ FF = 4\pi L^2/(4\pi L)^2 = 4\pi L^2/16L^2 = \pi/4 = 0.785. \]

The second feature needed to be evaluated is the Eccentricity, which is calculated as shown in the following equation:

\[ EC = c/a, \] (4)

where \( c \) is the distance between the center of the object and the focus and \( a \) is the major axis or the distance between the center and the vertex of the object. The larger the eccentricity of the object is, the less it likely to be a ball. For instance, if the object is a circle with radius \( r \) as in Figure 11A, it will be \( EC = c/a = 0/r = 0 \). This is because the center of the circle is also its focus and its major axis is the radius; thus, the eccentricity of a circle is zero. Finally, the third feature is the Area Ratio that is computed as in the following equation:

\[ AR = A_{Obj}/A_{MBR}, \] (5)

where \( A_{Obj} \) and \( A_{MBR} \) are the object area and its minimal bounding rectangle (MBR) area, respectively. For example, in case the object is a circle with radius \( r \) as in Figure 11A, the area of the object will be \( \pi r^2 \). For the bounding rectangle of the circle, it will be considered as a square with length \( 2r \); the area of the bounding square will be \( (2r)^2 = 4r^2 \). The Area Ratio of the circle will be \( AR = \pi r^2/4r^2 = \pi/4 = 0.785 \). Consequently, the optimal Area Ratio of a circle is 0.8.

As illustrated in the previous paragraphs, a candidate ball maybe a ball in case the Form Factor FF (BC) and the Area Ratio AR (BC) are high while the Eccentricity EC (BC) is low. On the other hand, the candidate may not be a ball when the form factor FF (BC) and the Area Ratio AR (BC) are low and the Eccentricity EC (BC) is high. In this paper, the previous three features can be employed to establish the BFIE, which consists of a group of fuzzy if-then rules with suitable membership functions to generate the specified input-output pairs. The membership functions and the fuzzy rules will be manually determined to model the outputs and explain the input–output relationship of the system. Generally, the architecture of BFIE includes three steps that have to be considered, and they are (i) Inputs Fuzzification, (ii) Fuzzy Rule Induction, and (iii) Defuzzification.
a. Inputs Fuzzification

The three features, which are FF (BC), AR (BC), and EC (BC), will be counted as the fuzzy sets of the BFIE. In the fuzzification step, a membership function for each used fuzzy set will be determined, which gives the similarity value for input according to each used fuzzy set. And then, the input values will be interpreted to the corresponding membership linguistic variables, which are “Low” and “High.” Such a process returns a value between 0.0 and 1.0 for non-membership and full-membership, respectively. The employed membership functions for the used fuzzy sets (e.g., FF, AR, and EC) are illustrated in Figure 12.

b. Fuzzy Rule Induction

In this step, a group of fuzzy rules, which are in the If-then patterns, are used to combine the fuzzified inputs from the former step. These rules can be formed as the following example if (x is A) AND (y is B) AND (z is C), THEN (w is D), where x, y, and z are the input values (e.g., FF, AR, and EC), while A, B, and C are the corresponding linguistic variables (e.g., low or high), thus, w is the rule output, and D is a linguistic variable (low or high). The "IF" part of this rule “x is A” is called the “antecedent” or “premise,” while the "THEN" part “w is D” is called the “consequence” or “conclusion.” In this paper, eight fuzzy rules will be used, which are listed in Table 2, where ‘L’ means “Low” and ‘H’ means “High.” For more explanation, the first rule in Table 2 indicates that IF FF is Low AND AR is Low AND EC is Low, THEN Output is Low.

Fuzzy rules can be integrated by a process called “Composition.” There are many common composition methods, which are max-product, max-min, drastic product, and sum-dot. In this

![Illustrated example shapes. (A) Circle and (B) square](http://wileyonlinelibrary.com)

![Input membership functions](http://wileyonlinelibrary.com)
paper, the max-min technique will be employed by finding the min value between all the premises of the rule, while the max value is used for aggregation. Hence, the premise and consequence are represented by the form of linguistic variables, the max-min inference rule can be illustrated in the following equation:

\[ \mu_{\text{out}}(x) = \frac{\text{aggregation}}{\text{MAX}} \left[ \min_{\text{implication}} (\mu_{\text{inp}(1)}, \mu_{\text{inp}(w)}) \right] \quad \forall x \in \{1, 2, 3, \ldots, q\}. \quad (6) \]

c. Defuzzification

In this step, the output from the former step has to be defuzzified to produce crisp numeric values, which will be suitable for real-time applications. The output of the BFIE will be also a fuzzy set. The most common defuzzification technique is the Center of Gravity (COG), which is used in this paper. In this technique, the average area under the curve of the membership function is calculated as in (7) to find the output crisp value. In our proposed paper, defuzzification is done using the output membership function illustrated in Figure 13. In case a Ball Candidate BC whose input parameters are FF(BC), AR(BC), and EC(EC); the crisp output value of the defuzzification step will be called the Similarity Value (SV) of the ball candidate BC, for example, SV(BC). Finally, the decision is taken whether ball candidate BC is the actual

| ID | FF | AR | EC | Rule output |
|----|----|----|----|-------------|
| 1  | L  | L  | L  | L           |
| 2  | L  | L  | H  | L           |
| 3  | L  | H  | L  | H           |
| 4  | L  | H  | H  | L           |
| 5  | H  | L  | L  | H           |
| 6  | H  | L  | H  | L           |
| 7  | H  | H  | L  | H           |
| 8  | H  | H  | H  | H           |

**Figure 13** The output membership function [Color figure can be viewed at wileyonlinelibrary.com]
ball or not according to a simple rule, which is represented by the simple step identification function illustrated in Figure 14.

\[ COG = \frac{\sum \mu(SV) \times SV}{\mu(SV)}. \]  

(7)

5.3 | Ball detection optimization: Illustrative example

In this section, an example of ball candidate classification will be illustrated and explained in detail in Figure 15. For example, a ball candidate BC1 that has FF = 0.6, AR = 1 and EC = 1.4 will be classified as a ball or not in the following steps. These steps are the three steps of the Fuzzy Inference System that is used to decide whether the candidate is a ball or not. In the first step, the fuzzy set membership values for this ball candidate will be determined as shown in Figure 15 Step (1). These values will be \( \mu_L(FF = 0.6) = 0.7, \mu_H(FF = 0.6) = 0.3, \mu_L(AR = 0.6) = 0.4, \mu_H(AR = 0.6) = 0.6, \mu_L(EC = 0.6) = 0.8, \mu_H(EC = 0.6) = 0.2 \) for low and high membership values of Form Factor, Area Ratio, and Eccentricity fuzzy sets.

In the second step, the fuzzy rules will be applied to the membership values from the previous step. As shown in Figure 15 Step (2), all the expected fuzzy outputs for each rule will be determined, and then the max-min method will be used to calculate the output values of the fuzzy rules. Therefore, the output will be High with membership = \( \text{Output}_H = \text{MAX}(0.6, 0.3, 0.3, 0.2) = 0.6 \) and Low with membership = \( \text{Output}_L = \text{MAX}(0.4, 0.2, 0.2, 0.2) = 0.4 \). In the last step, these output values will be defuzzified using the output membership function as in Figure 15 Step (3). The weighted average of the area bounded by the membership function curve is computed using the COG method as in (7). This result crisp value will identify the Similarity Value of the ball candidate BC1 that will be 9.05. From Figure 14, if the crisp value is greater than \( \beta \), the candidate ball will be a ball and vice versa. Consequently, the Similarity Value is greater than the assumed threshold \( \beta = 7 \), and then the given candidate ball BC1 will be a ball. \( COG = \sum_{R=1}^{4} R \times 0.4 + \sum_{R=5}^{16} R \times 0.6/4 \times (0.4) + 12 \times (0.6) = 9.05 \).

6 | PERFORMANCE ANALYSIS AND IMPLEMENTATION

In this section, the proposed FB^2D strategy will be clearly evaluated. In this paper, Table 3 illustrates the adjustable parameters that will be employed with their corresponding values.
In this section, six strategies will be compared based on various metrics, which aim to determine which of these strategies can perfectly detect the ball. The notations, shown in Table 4, are the outcomes of the system used for the evaluation of the performance of each algorithm. In

### Table 3: The used tunable parameters

| Parameter               | description                                                      |
|-------------------------|------------------------------------------------------------------|
| Bee height              | The Bee altitude (height) from the ground.                       |
| Bee window size         | The coverage area of the Bee, which depends on the Bee height.   |
| Bee speed               | The speed of the Bee                                             |
| Bee camera resolution   | The resolution of the image taken by the camera.                 |

### 6.1 Performance metrics

In this section, six strategies will be compared based on various metrics, which aim to determine which of these strategies can perfectly detect the ball. The notations, shown in Table 4, are the outcomes of the system used for the evaluation of the performance of each algorithm. In
In this paper, the different used metrics are precision, recall, accuracy, error, and F1 scores as in (8), (9), (10), (11), and (13), respectively.

\[
\text{Precision} = P = \frac{A}{A + B}, \quad (8)
\]
\[
\text{Recall} = R = \frac{A}{A + D}, \quad (9)
\]
\[
\text{Accuracy} = Acc = \frac{A + C}{A + B + C + D}, \quad (10)
\]
\[
\text{Error} = E = \frac{B + D}{A + B + C + D}. \quad (11)
\]

In this paper, Precision (P) and Recall (R) can be combined in one metric called F-measure, which can be calculated as in the following equation:

\[
F_n = \frac{(n^2 + 1) \times P \times R}{n^2 \times P + R}, \quad (12)
\]

where \(n\) is a weighting factor, which can set various values to Precision (P) and Recall (R). The most commonly used F-measure is \(F_1\) as \(n = 1\) for both precision and recall, which can be as in the following equation:

\[
F_1 = \frac{2 \times P \times R}{P + R}. \quad (13)
\]

### 6.2 Competitors and the evaluation image sequences

Experiments were carried out on different image sequences, which are taken in matches played in FIFA World Cup Russia 2018. Our algorithm has been compared to six different algorithms, as shown in Table 5; they are EBBE algorithm, a deep learning approach for 2D ball detection and tracking (DLBT) in soccer videos, a deep network-based object detector specialized for ball detection in long shot videos (DeepBall), Improved Non-Maximum Suppression (INMS), Cuckoo Search Algorithm and a real-time ball detection approach based on the YOLOv3 object detection model (micro YOLOv3). Supposing that the camera of the ball bee is at a fixed height, the ball will correspond to a circular region with radii in the range \(R_{\text{MIN}} = 24, \ldots\)

| Outcome | Description                          |
|---------|--------------------------------------|
| A       | Number of frames where the ball is detected correctly |
| B       | Number of frames where the ball is detected incorrectly |
| C       | Number of frames where the ball is undetected correctly |
| D       | Number of frames where the ball is undetected incorrectly |
$R_{\text{MAX}} = 28$) depending on the distance of the ball with respect to the optical center of the camera. The whole image size was $1280 \times 720$ pixels for every image in the used sequences. Ten image sequences have been considered in our experiments, which have been taken with artificial light and the ball appears as a complete circular shape. Each sequence contains 100 images divided into 50 images that contain the complete ball and the other 50 images, which contain no ball.

| Classification technique                                                                 | Description                                                                                                                                                                                                 |
|----------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Entropy-Based Initial Background Extraction (EBBE) Algorithm\(^{33}\)                   | It was used to detect the initial backgrounds. First, through a color distance classifier. Second, the concept of color category entropy is proposed. Third, when the background is masked for a long time, the initial-background convergence time can be dynamically determined via the magnitudes of color category entropy to prevent erroneous detection results due to moving objects being mistaken for a background. |
| a deep learning approach for 2D ball detection and tracking (DLBT) in soccer videos\(^{43}\) | A new 2-stage buffer median filtering background modeling is used for moving objects blob detection. A deep learning approach for classification of an image patch into three classes, i.e., ball, player, and background are initially proposed. |
| A deep network-based object detector specialized for ball detection in long shot videos (DeepBall)\(^{44}\) | The network uses hypercolumn concept, where feature maps from different hierarchy levels of the deep convolutional network are combined and jointly fed to the convolutional classification layer. |
| Improved Non-Maximum Suppression (INMS)\(^{34}\)                                       | INMS is used to detect the multiple objects in less time with higher accuracy. The main objective of the proposed model is to address as many challenges as possible along with detection of multiple objects in a single image. |
| Cuckoo Search Algorithm\(^{35}\)                                                       | In this algorithm, the fitness function evaluation is started by calculating the number of instances in total of each agent $\alpha$ and it also calculates the selection of number of instances in each agent $\beta$ and the selection of number of boundary instances selected by agent $\gamma$. The boundary instances are selected by the algorithm and they are calculated and used for the evaluation of fitness value. |
| A real-time ball detection approach based on the YOLOv3 object detection model (micro YOLOv3)\(^{45}\) | It applies specific adjustments to the network architecture and training process to enhance the detection accuracy and speed: We facilitate an efficient integration of motion information, avoiding a complex modification of the network architecture. |
6.3 Evaluating the proposed FB$^2$D strategy

Our proposed FB$^2$D strategy will be compared to the other strategies illustrated in Section 6.2 and Table 5. FB$^2$D succeeds in detecting the ball, whereas the other strategies may fail to detect it. The experimental results are illustrated in Figures 16–19; they present the precision, recall, accuracy, error, and F1-measure, respectively. Our strategy achieves the higher values for precision, recall, accuracy, and F-measure in comparison to the other strategies. On the other hand, it has the lowest error rate compared to the others. Our proposed algorithm’s results of precision, recall, accuracy, and F1-measure for all the used ten image sequences, as shown in Table 6.

Figure 16 shows the precision for all the 10 sequences for the 5th sequence; the precisions are 0.98, 0.79, 0.91, 0.93, 0.84, 0.88, and 0.95 for our FB$^2$D, EBBE, DLBT, DeepBall, INMS, Cuckoo and micro YOLOv3, respectively. However, in the 6th sequence, the values are 0.9, 0.7, 0.83, 0.85, 0.74, 0.79, and 0.87 for FB$^2$D, EBBE, DLBT, DeepBall, INMS, Cuckoo, and micro YOLOv3, respectively. These values are different as the ball distinctly appears in the positive images of the fifth sequence. But there are blur images in the sixth sequence, which made the
existence of the ball not clear. Also, the recall illustrated in Figure 17, its values in the 2nd sequence are 0.96, 0.87, 0.88, 0.9, 0.78, 0.82 and 0.91 for FB2D, EBBE, DLBT, DeepBall, INMS, Cuckoo and micro YOLOv3, respectively. In the 9th sequence, the recall values are 0.89, 0.83, 0.65, 0.68, 0.82, 0.8, and 0.72 for FB2D, EBBE, DLBT, DeepBall, INMS, Cuckoo, and micro YOLOv3, respectively. The recall values are in this form because the ball does not exist obviously in the negative images of the ninth sequence. On the other hand, the ball does not appear clearly in the positive images of the second sequence.

Respectively, Figures 18 and 19 show both the accuracy and the error of our proposed algorithm in comparison to the others used by some competitors. For example, in the sequence number one, the accuracies are 0.9, 0.77, 0.79, 0.8, 0.7, 0.73, and 0.81, while the error values are 0.1, 0.23, 0.21, 0.2, 0.3, 0.27, and 0.19 for FB2D, EBBE, DLBT, DeepBall, INMS, Cuckoo, and micro YOLOv3, respectively. In sequence number 5, the accuracy values are 0.96, 0.88, 0.89, 0.9, 0.78, 0.82, and 0.92, but the error values are 0.04, 0.12, 0.11, 0.1, 0.22, 0.18, and 0.08 for FB2D, EBBE, DLBT, DeepBall, INMS, Cuckoo, and micro YOLOv3, respectively. Finally, the F1
measure values have been shown in Figure 20; in the 5th sequence, these values are 0.96, 0.81, 0.89, 0.87, 0.77, 0.83, and 0.93 for FB2D, EBBE, DLBT, DeepBall, INMS, Cuckoo and micro YOLOv3, respectively. Furthermore, in the 6th sequence, the F1 values are 0.9, 0.73, 0.82, 0.82, 0.69, 0.74, and 0.86 for FB2D, EBBE, DLBT, DeepBall, INMS, Cuckoo, and micro YOLOv3, respectively. As the difficulty of the ten sequences is varied, the results are different. For example, the negative images of the fifth sequence have more difficult cases, so the values of the accuracy are lower than their values in the first sequence.

In our proposed paper, the average values for each of the precision, recall, accuracy, error, and F1-measure for 3, 5, 8, and 10 sequences will be computed using our proposed strategy FB2D. These results prove that the FB2D system is more effective and detects the ball perfectly. Figure 21 illustrates the results for all the used sequences, which are 0.93, 0.92, 0.93, 0.07, and 0.93 for average precision, average recall, average accuracy, average error, and average F1-measure, respectively when only three sequences are used. On the other hand, when five sequences are used; average precision, average recall, average accuracy, average error, and average F1-measure are 0.94, 0.92, 0.93, 0.07, and 0.93, respectively. In the case of using eight sequences, average precision, average recall, average accuracy, average error, and average F1-measure are 0.94, 0.92, 0.93, 0.07, and 0.93 respectively. Finally, when the whole ten sequences are used; average precision, average recall, average accuracy, average error, and average F1-measure are 0.95, 0.92, 0.93, 0.07, and 0.93, respectively.

The experiments under study were conducted on various balls of different colors as shown in Figure 22. The proposed FB2D strategy can successfully detect balls with various colors. Hence, the results of the suggested algorithm, shown in Figures 16–20, prove that all balls are located in different positions on the field. The frame with the fast-moving ball suffers from motion blurring, which will make this ball larger and will reduce its white tone than the slower-moving or stationary ball. The white color percentage within the candidate ball must never be lower than 30% of the entire region.46

FIGURE 20 F1 Measure of all algorithms for the 10 sequences [Color figure can be viewed at wileyonlinelibrary.com]

FIGURE 21 Average precision, recall, accuracy, error and F1-measure for 3, 5, 8, and 10 sequences [Color figure can be viewed at wileyonlinelibrary.com]
Finally, the experimental results prove that the performance values of our proposed strategy exceed the other competitors' strategies with respect to precision, recall, and accuracy. There are three reasons for these high results: the first one is that the ball always has a circular shape because the camera is fixed orthogonally under the flying bee. While, the camera in the other competitors' strategies is static and faces only one side of the playfield, which makes the ball has distorted patterns. This reason makes it not easy to detect the ball professionally as it may have several patterns depends on the projection angle of the camera. The second reason is that our flying bees effectively cover the whole area of the playground. Furthermore, the majority of the traditional detection strategies have employed static cameras, which need complex zooming and calibration methods to perform ball detection efficiently. Also, strategies based on moving cameras still cannot efficiently detect the ball due to the randomness movement of the ball. The third and final reason is that FB2D relies on simple computations in comparison to the traditional detection systems, and it guarantees a fast response of the bee. Besides, there is no delay in identifying the ball as fog computing is employed. Consequently, our detection algorithm is suitable for real-time detection systems.

7 | FB2D PROS AND CONS

In this section, different pros and cons of the proposed FB2D strategy will be discussed. As depicted in Table 7, FB2D is a robust technique employing fuzzy logic, which is not very sensitive to changing environments and erroneous or forgotten rules. The proposed FB2D is also reliable because it relies on the engagement between the used ball detection technique and the hosting ball bee in a mutually beneficial relationship. Our proposed system is efficient in detecting the playfield color as it constructs an accurate range of green that represents the playing field. Also, the background model is continuously updated using the Scanner Bee. Thus, FB2D has the capability of online training and adaptive response. Besides, rules in the fuzzy logic system can be added and deleted due to the flexibility of fuzzy logic. In FB2D, the background modeling is done offline. Furthermore, any update in the background model is done in parallel during the system operation, which allows the system to be fast and consistent with the ball detection task. In FB2D, fewer values, rules, and decisions are
| **FB²D Pros** | **Description** |
|----------------|-----------------|
| Efficient      | * FB²D is efficient in detecting the playfield color as it constructs accurate range of green that represents the playing field. |
|                | * FB²D employs an effective fuzzy inference engine (FIE) which pretends the way in which humans interpret linguistic values. |
|                | * The employed BFIS relates output to input, without having to understand all the variables, permitting the design of a system to be more accurate and stable. |
|                | * FB²D has a fully convolutional architecture processing entire image at once in a single pass. This is much more computationally effective than the sliding window approach introduced in [4]. |
| Adaptive       | * The background model is updated continuously by updating the H_Vector using the scanner bee. Thus, FB²D has the capability of on-line training and adaptive response. |
| Robust         | FB²D is a robust system as it employs fuzzy logic so that: |
|                | * No precise inputs are required. |
|                | * Fuzzy algorithms are often robust, in the sense that they are not very sensitive to changing environments and erroneous or forgotten rules. |
| Flexibility    | * A fuzzy logic system is flexible and allow easily modification in the rules. |
|                | * Due to the flexibility of fuzzy logic, we can add and delete rules in FLS system. |
|                | * FB²D can be successfully employed to detect different kinds of balls not only the soccer ball. |

| **FB²D Cons** | **Description** |
|----------------|-----------------|
| Property       | Description     |
| Initialization time | Need some time before the match starts to detect the playfield color. |
| Expert guidance | * Fuzzy logic should be built with the complete guidance of experts. |
|                | * Setting exact, fuzzy rules and, membership functions is a difficult task |
| Occlusion      | FB²D has problems to deal with the ball when it is partially occluded. |

(Continues)
| FB²D Pros | FB²D Cons |
|-----------|-----------|
| **Property** | **Description** |
| Fast | • In FB²D, the reasoning process is simple, compared to computationally precise systems, so computing power is saved. This is a very interesting feature, especially in ball detection as a real time system.  
• Fuzzy methods have a shorter development time than conventional methods.  
• In FB²D, the background modeling is done offline. Furthermore, any update in the background model is done in parallel during the system operation which allows the system to be fast and consistent with ball detection task. |
| Simplicity | • The structure of Fuzzy Logic System, which is the core of the proposed FB²D, is easy and understandable.  
• In FB²D, fewer values, rules, and decisions are required.  
• FB²D can be easily constructed and deployed. |
| Reliable | • FB²D is reliable due to the engagement between the used ball detection technique and the hosting ball bee in a mutually beneficial relationship. |
required, so the fuzzy logic system can be easily constructed and deployed. On the other hand, the proposed FB²D suffers from some drawbacks, and they are as follows: (i) initialization time, (ii) expert guidance, and (iii) occlusion. Concerning the first one, the system needs some time before the match starts to detect the playfield color. In the second drawback, fuzzy logic should be built with the complete guidance of experts. For the third one, our system has problems dealing with the ball when it is partially occluded. Details about the pros and cons of FB²D are illustrated in Table 7.

8 | CONCLUSION

This paper introduced a strategy for soccer ball detection using a Ball Bee. Our proposed FB²D Strategy is divided into two main phases: (a) background identification, and (b) object detection. In the first phase, the HSV color space had been used to find the hue values of the empty field before the game started. And then, these values are compared to the corresponding values in the current frame of interest. On the other hand, the second phase consists of four subphases: (i) background subtraction, (ii) edge detection, (iii) ball candidates identification, and (iv) ball fuzzy inference system (BFIS). In the first subphase, the matching hue values of the taken frame will be removed, and then the Canny edge detector has been applied in the second subphase. In the third subphase, the candidate ball from the objects found in the frame has been identified using the Circle Hough Transform technique. Finally, in the fourth subphase, a ball fuzzy inference engine has been employed with three fuzzy sets and eight rules to decide which candidate is the actual ball according to specific membership functions. Experimental results illustrate that soccer ball detection in real-time applications can be done even if the ball is occluded by other objects or players. The system aims to help professional football analysts to evaluate the players' performance, by allowing automatic indexing and retrieval of interesting events. In spite of its effectiveness, the proposed FB²D consumes some time before the match starts, as the Scanner Bee takes a few frames of the empty playfield to determine its true color values. It also has problems detecting the ball when it's possessed or partially occluded by a player.

ACKNOWLEDGMENTS

The authors gratefully thank the Editor-in-Chief and two anonymous referees for their valuable and constructive comments, which have much improved the paper.

CONFLICT OF INTERESTS

The authors declare that there are no conflict of interests.

ORCID

Arwa E. Abulwafa [http://orcid.org/0000-0001-8998-2984]
Ahmed I. Saleh [http://orcid.org/0000-0003-4141-9257]
Mohamed S. Saraya [http://orcid.org/0000-0001-5682-8243]
Hesham A. Ali [http://orcid.org/0000-0001-6675-7987]

REFERENCES

1. Upendra RM, Pati UC. A novel algorithm for detection of soccer ball and player. 2015 Int Conf Signal Process Commun (ICCCSP). 2015;13:344-348. https://doi.org/10.1109/icccsp.2015.7322903
2. Gomez G, Herrera LP, Link D, Eskofier B. Tracking of ball and players in beach volleyball videos. PLoS One. 2014;9(11):111730. https://doi.org/10.1371/journal.pone.0111730
3. Yoon Y, Hwang H, Choi Y, et al. Analyzing basketball movements and pass relationships using realtime object tracking techniques based on deep learning. *IEEE Access*. 2019;7:56564-56576. https://doi.org/10.1109/access.2019.2913953

4. Reno V, Mosca N, Marani R, Nitti M, D’Orazio T, Stella E. Convolutional neural networks based ball detection in tennis games. 2018 IEEE/CVF Conf Comput Vis Pattern Recognit Workshops (CVPRW). 2018;13:549-556. https://doi.org/10.1109/cvprw.2018.00228

5. Pobar M, Ivasic-Kos M. Active player detection in handball scenes based on activity measures. *Sensors*. 2020;20(5):1475. https://doi.org/10.3390/s20051475

6. Velastin SA, Fernández R, Espinosa JE, Bay A. Detecting, tracking and counting people getting on/off a metropolitan train using a standard video camera. *Sensors*. 20(21), 2020:6251.

7. Cuevas C, Quilón D, García N. Techniques and applications for soccer video analysis: a survey. *Multimed Tools Appl*. 2020;79(39-40):29685-29721.

8. Hansen P, Franco P, Kim SY. Soccer ball recognition and distance prediction using fuzzy Petri nets. 2018 IEEE Int. Conf. Inf. Reuse Integr. (IRI). IEEE. July 2018:6-9:315-322. https://doi.org/10.1109/IRI.2018.00055

9. Meneghetti DDR, Homem TPD, de Oliveira JHR, et al. Detecting soccer balls with reduced neural networks: a comparison of multiple architectures under constrained hardware scenarios. arXiv [csCV]. 2020. http://arxiv.org/abs/2009.13684

10. Zhou B, Koerger H, Wirth M, et al. Smart soccer shoe: monitoring foot-ball interaction with shoe integrated textile pressure sensor matrix. Proc. of the 2016 ACM Int. Symp. Wearable Comput. 2016:4-71. https://doi.org/10.1145/2971763.2971784

11. Heryanto MA, Suprijono H, Suprapto BY, Kusumoputro B. Attitude and altitude control of a quadcopter using neural network based direct inverse control scheme. *Adv Sci Lett*. 2017;23(5):4060-4064. https://doi.org/10.1166/asl.2017.8328

12. Abulwafa AE, Saleh AI, Ali HA, Saraya MS. A fog-based ball tracking (FB2T) system using intelligent ball bees. *J Amb Intell Hum Comp*. 2020;11(11):5735-5754. https://doi.org/10.1007/s12652-020-01948-6

13. Joshi N. 4 sensors that are being used in drones: IOT:Drones technology. Application development. 2016. Accessed February 1, 2020. https://www.allerin.com/blog/4-sensors-that-are-being-used-in-drones-technology

14. Brown L. What is a drone and how does it work. 2019. Accessed February 1, 2020. https://filmora.wondershare.com/drones/what-is-drone-how-does-it-work.html

15. Ishii N, Kitahara I, Kameda Y, Ohta Y. 3D tracking of a soccer ball using two synchronized cameras. *Adv Multimed Inf Process PCM*. 2007:196-205. https://doi.org/10.1007/978-3-540-77255-2_22

16. Ren J, Orwell J, Jones GA, Xu M. Tracking the soccer ball using multiple fixed cameras. *Comput Vis Image Underst*. 2009;113(5):633-642. https://doi.org/10.1016/j.cviu.2008.01.007

17. Polceanu M, Petac AO, Lebsir HB, Fiter B, Buche C. Real time tennis match tracking with low-cost equipment. 31th Int Fla. AI. Res. Soc. Conf. 2018:31:197-200.

18. Yan F, Christmas W, Kittler J. A tennis ball tracking algorithm for automatic annotation of tennis match. Proceedings of the British Machine Vision Conference BMVC. 2005;2:619-628. https://doi.org/10.5244/c.19.67

19. Owens N, Harris C, Stennett C. Hawk-eye tennis system. International Conference on Visual Information Engineering (VIE 2003) Ideas, Applications, Experience. University of Surrey, Guildford:182-185. https://doi.org/10.1049/cp:20030517

20. Conaire C, Kelly F, Connaghan D, O’Connor NE. TennisSense: A platform for extracting semantic information from multi-camera tennis data. 2009 16th International Conference on Digital Signal Processing, DSP. 2009:1-6. https://doi.org/10.1109/icdsp.2009.5201152

21. Fazio M, Fisher KS, Fujinami T. *Tennis Ball Tracking: 3D Trajectory Estimation Using Smartphone Videos*. Stanford, California: Department of Electrical Engineering, Stanford University; 2018.

22. Archana M, Geetha MK. Object detection and tracking based on trajectory in broadcast tennis video. *Procedia Comput Sci*. 2015;58:225-232. https://doi.org/10.1016/j.procs.2015.08.060

23. Teachabarikiti K, Chalidabhongse TH, Thammano A. Players tracking and ball detection for an automatic tennis video annotation. 2010 11th International Conference on Control Automation Robotics & Vision. ICRA; 2010:2491-2494. https://doi.org/10.1109/icarcv.2010.5707906

24. Huang MY, Huang CP. An optimal image segmentation clustering algorithm for badminton sport moving tracking applications from video sequences. *Int J Inf Secur (IJCSIS)*. xi. 2017;6(9):207-219.
25. Wang JR, Parameswaran N. Survey of sports video analysis: research issues and applications. Proceedings of the Pan-Sydney area workshop on Visual information processing. 2004;36:87-90.

26. Fu Z, Han Y. Centroid weighted Kalman filter for visual object tracking. Measurement. 2012;45(4):650-655. https://doi.org/10.1016/j.measurement.2012.01.004

27. Yao L, Ling M. An improved mixture-of-Gaussians background model with frame difference and blob tracking in video stream. Sci World J. 2014;2014:41-9. https://doi.org/10.1155/2014/774088.1774295

28. Kumar S, Yadav JS. Background subtraction method for object detection and tracking. Proceeding of International Conference on Intelligent Communication, Control and Devices, ICICCD. 2016:1057-1063. https://doi.org/10.1007/978-981-10-1708-7_125

29. Vo GD, Park C. Robust regression for image binarization under heavy noise and nonuniform background. Pattern Recognit. 2018;81:224-239. https://doi.org/10.1016/j.patcog.2018.04.005

30. Cheng FC, Huang SC, Ruan SJ. Advanced motion detection for intelligent video surveillance systems. Proc. ACM Symp. Appl. Comput. (SAC ’10). 2010;15:983-984. https://doi.org/10.1145/1774088.1774295

31. D’Orazio T, Guaragnella C, Leo M, Distante A. A new algorithm for ball recognition using circle Hough transform and neural classifier. Pattern Recognit. 2004;37(3):393-408. https://doi.org/10.1016/s0031-3203(03)00228-0

32. Moyyila U. Detection and Recognition of Soccer Ball and Players. PhD dissertation; 2015.

33. Chiu SY, Chiu CC, Xu S. A background subtraction algorithm in complex environments based on category entropy analysis. Appl Sci. 2018;8(6):885. https://doi.org/10.3390/app8060885

34. Bondalapati A, Bhavanan SN, Reddy ES. Moving object detection based on unified model. J Amb Intel Hum Comp. 2020;12(6):6057-6072. https://doi.org/10.1007/s12652-020-02170-0

35. Balaji SR, Karthikeyan S, Manikandan R. Object detection using Metaheuristic algorithm for volley ball sports application. J Amb Intel Hum Comp. 2020;12(1):375-385. https://doi.org/10.1007/s12652-020-01981-5

36. Huang Y, Llach J, Bhagavathy S. Players and ball detection in soccer videos based on color segmentation and shape analysis. Multimedia Content Analysis and Mining (MCAM). 2007;416-425. https://doi.org/10.1007/978-3-540-73417-8_50

37. Ali MN, Abdullah-Al-Wadud M, Lee SL. An efficient algorithm for detection of soccer ball and players Proc. 16th ASTL Control and Networking. 2012;16:1-8.

38. Mudjirahardjo P, Purnomo MF, Hasanah RN, Suyono H. Multi scale performance of feature extraction for human head recognition. Int J Appl Eng Res. 2015;10(17):37731-37736.

39. Mudjirahardjo P, Edy Purnomo MF, Nur Hasanah R, Suyono H. Histogram of transition for human head recognition. J Teknol. 2016;78(5-9). https://doi.org/10.11113/jt.v78.8787

40. Chakraborty B, Meher S. A trajectory-based ball detection and tracking system with applications to shooting angle and velocity estimation in basketball videos. Proc. 2013 Annu. IEEE India Conf. (INDICON). 2013;13:1-16. https://doi.org/10.1109/indicon.2013.6725963

41. Durus M. Ball Tracking and Action Recognition of Soccer Player in TV Broadcast Videos. PhD Dissertation Technische Universitat Munchen; 2014.

42. Mudjirahardjo P, Tan J, Kim H, Ishikawa S. Abnormal motion detection in an occlusive environment. Proceeding of SICE Annual ConferenceNagoya-Japan; 2013:1398-1402. https://doi.org/10.13140/2.1.5108.9284

43. Kamble PR, Keskar AG, Bhurchandi KM. A deep learning ball tracking system in soccer videos. Opto-Electron Rev. 2019;27(1):58-69. https://doi.org/10.1016/j.operev.2019.02.003

44. Komorowski J, Kurzejamski G, Sarwas G. DeepBall: Deep neural-network ball detector. Proc. 14th Int. Jt. Conf. Comput. Vis. Imaging Comput. Graph. Theory Appl. 2019:297-304. https://doi.org/10.5220/0007348902970304

45. Hiemann A, Kautz T, Zottmann T, Hlawitschka M. Enhancement of speed and accuracy trade-off for sports ball detection in videos-finding fast moving. Small Objects Real Time Sens. 2021;21(9):3214. https://doi.org/10.3390/s21093214

46. Ren J, Xu M, Orwell J, Jones GA. Real-time modeling of 3-D soccer ball trajectories from multiple fixed cameras. IEEE Trans Circuits Syst Video Technol. 2008;18(3):350-362. https://doi.org/10.1109/tcsvt.2008.918276

47. Monnet A, Mittal A, Paragios N, Ramesh V. Background modeling and subtraction of dynamic scenes. Proc. Ninth IEEE Int. Conf. Comput. Vis. 2003;13:1305-1312. https://doi.org/10.1109/icip.2003.1238641

48. Kirti R, Bhatnagar A. Image segmentation using Canny edge detection technique. Proc. of Int. J. Technol. Manag. Resear. March 2017: 8-14.
49. Minematsu T, Uchiyama H, Shimada A, Nagahara H, Taniguchi RI. Adaptive background model registration for moving cameras. Pattern Recognit Lett. 2017;96:86-95. https://doi.org/10.1016/j.patrec.2017.03.010

50. Chen W, Shi YQ, Xuan G. Identifying Computer Graphics using HSV Color Model and Statistical Moments of Characteristic Functions. Proc IEEE Int. Conf. Multimed. Expo. 2007;13:1123-1126. https://doi.org/10.1109/icme.2007.4284852

51. Pitas I. Digital Image Processing Algorithms and Applications. New York: John Wiley & Sons (JWS); 2000.

52. Vinutha HP, Poornima B, Sagar BM. Detection of outliers using interquartile range technique from intrusion dataset. Adv Intell Syst Comput. 2018;701:511-518. https://doi.org/10.1007/978-981-10-7563-6_53

53. Canny J. A computational approach to edge detection. IEEE Trans Pattern Anal Mach Intell. 1986;8(6):679-698.

54. Yadav VK, Batham S, Acharya AK, Paul R. Approach to accurate circle detection: circular Hough transform and local maxima concept. Int. Conf. Electron. Commun. Comput. Sys. (ICECS). 2014;13:1-5. https://doi.org/10.1109/ecs.2014.6892577

How to cite this article: Abulwafa AE, Saleh AI, Saraya MS, Ali HA. A new ball detection strategy for enhancing the performance of ball bees based on fuzzy inference engine. Int J Intell Syst. 2022;37:9620-9654. https://doi.org/10.1002/int.22681