

A New Ball Detection Strategy for Enhancing the Performance of Ball Bees Based on Fuzzy Inference Engine

Arwa Abulwafa (✉ aeabulwafa@gmail.com)

Mansoura University Faculty of Engineering

Ahmed I. Saleh

Mansoura University Faculty of Engineering

Mohamed S. Saraya

Mansoura University Faculty of Engineering

Hesham A. Ali

Mansoura University Faculty of Engineering

Research Article

Keywords: Background Subtraction, Edge Detection, Soccer, Ball Detection, Quadcopter.

Posted Date: May 17th, 2021

DOI: <https://doi.org/10.21203/rs.3.rs-286620/v1>

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

[Read Full License](#)

Version of Record: A version of this preprint was published at International Journal of Intelligent Systems on September 19th, 2021. See the published version at <https://doi.org/10.1002/int.22681>.

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 and Hesham A. Ali
Dept. of Computer Eng. & Systems, Faculty of Engineering, Mansoura University, Egypt.

Abstract

Sports video analysis has received much attention as it is turned to be a hot research area in the field of image processing. This led to opportunities to develop fascinating applications supported by analysis of different sports especially football. Identifying the ball in soccer images is an essential task for not only goal scoring but also players' evaluation. However, soccer 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 etc., ball detection in soccer games is still an open research area. In this paper, a new Fuzzy Based Ball Detection (FB²D) strategy is proposed for identifying the ball through a set of image sequences extracted from a soccer match video. FB²D has the ability to accurately identify the ball even if it is attached to the white lines drawn on the playground or partially occluded behind players. FB²D has been compared to recent ball detection techniques. Experimental results have shown that FB²D outperforms recent detection techniques as it introduced the maximum accuracy and the accuracy of detection in the testing stage is close to 100%. As well as the minimum error.

Keywords: Background Subtraction, Edge Detection, Soccer, Ball Detection, Quadcopter.

1. Introduction

Detect and track the moving objects are key tasks for Artificial Intelligence (AI). Accordingly, in order to succeed in and thrive in this niche, there must exist an accurate technique for the detection of moving objects. On the other hand, sport is equivalent to motion in which people and objects are moving in different speeds and directions. In the case of ball-based sports such as soccer [1], volley [2], basketball [3], tennis [4] and hand ball [5], the ball is always the main center of attention. If the ball can be accurately detected even at high speeds, it will be easy to detect the key events of the game, metrics and so much more useful information. Moreover, an ability to accurately detect and track the ball in a video sequence is a core capability of an AI system aiming to automate analysis of matches as well as players' progress. Recently, there are several available techniques for detecting human bodies [6], hence players will be easy to detect but not so the ball. As a result, there is an urgent need to suggest robust techniques for detecting the moving ball. However, only being able to detect the ball won't be sufficient. It is believed that a good ball detector must be efficient and fast. Late detection of the ball will not be useful as it will not be possible to take correct decisions or actions based on this detection, simply because such action or decision will be useless.

As soccer is one of the most popular sports all over the world, the automatic analysis of its videos [7] has become a focal point of the research efforts. Their possible applications have broad ranges (e.g., indexing, retrieval, annotation, summarization, event detection, and tactics analysis). For the present, playfield and ball detection [8] always draw much attention since a playfield is the place, where almost all the events of the game take place in and the ball is the focus of both audience and players.

Throughout the soccer, as any ball game, players are continuously interacting with the ball. The ball remains in focus of attention during the whole time of the game as most actions take place around it. Therefore,

providing accurate ball detection, which is essential for ball tracking, is a necessary for comprehensive and deeper analysis of soccer games. Despite the higher quality hardware power and imaging devices, it might still be difficult to accurately detect the ball within the frame of interest. For accurate ball detection, two issues should be considered, which are; (i) accurate ball detection mechanism should be employed to precisely detect the ball, and (ii) it is important to not only get high quality images, but also to have images with explicit ball appearance as much as possible (e.g., the ball is not occluded behind players bodies). For the first issue, different soft computing techniques such as fuzzy [9] and neural networks [10] could be employed to promote the performance of the detection system. On the other hand, to resolve the second issue, interactive photographing [11] should be used in which the camera is movable to allow continuous ball tracking. In interactive photographing the object of interest (e.g., the ball) is under focus all the time. Accordingly, there is a need for a camera that continuously tracks the ball. This can be accomplished by using ball bees.

A ball bee is a special type of quadcopters [12]; it has the ability to track a moving ball. It was firstly introduced by Abulwafa et al. in [13] and used to track the moving ball in soccer matches. A ball bee is considered as 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 so many working sensors on board (e.g., speed and distance sensors, infrared and thermal sensors, image sensors, chemical sensors, GPS, and etc.) [14, 15]. In fact, there exists an engagement between the used ball detection technique and the hosting ball bee in a mutually beneficial relationship. To narrate confidently, the employed ball bee is used as 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 accurate ball detection technique will promotes 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 in order 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) normally, the ball has spherical shape but it might be observed as blurred due to its rapid motions and unpredictable acceleration, additionally, 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., (iv) 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 exits the camera field of view, (vi) it is difficult to detect the playfield due to the shadow brought out by illumination sunlight, (vii) The ball's features (e.g., color, shape, and size) are changing ones with the circumstance conditions like light and velocity, (viii) It is difficult to segment the ball from the player when players possess the ball, and (ix) the fragments due to inappropriate segmentation of players or field-lines 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 [16] while the others employ multiple and mostly dedicated cameras (e.g., offside cameras) [17]. The most successful techniques in 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 really contains the ball.

Candidate ball extraction can be performed using global information such as color, size, and shape or a combination of them. Particularly, the Circular Hough transform (CHT) has been recognized as a robust technique for curve detection [18]. On the other hand, which procedure is the best for validating ball candidates is more controversial. In [19], a candidate verification technique based on Kalman filter is presented. In [17], ball color, size, velocity and longevity are used to discriminate the ball from other frame

objects. However, these techniques experience several hurdles in ball candidate validation especially when the ball abruptly changes its trajectory. Other techniques focus on ball pattern extraction and recognition. For illustration, in [20] a neural architecture for discriminating the wavelet coefficients extracted from ball and no-ball examples is introduced. In spite of its efficiency, this technique fails to validate ball candidates in occluded balls since wavelet description requires a huge visible ball portion to supply accurate coefficients. In [21], a comparative study of the most recent strategies for ball detection was introduced. The study demonstrated that none of the techniques in literature can efficiently cope with all the possible variations of the ball's appearance. Moreover, authors concluded that; the problem is opened for more enhancements as it still requires more investigations

This paper introduces a new Fuzzy Based Ball Detection (FB²D) strategy for the detection of soccer ball using Ball Bees. FB²D consists of two main phases, which are (i) Background Identification and (ii) Object Detection. The main target of the first phase, which is done offline before starting the match, is to detect the background model. Although the dominant color of the background is the green, it is needed to identify the exact greening range of the playground. The intensity and degree of greening of the playground varies according to the color of the grass that 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, if the match takes place during the day, or according to the distance from the lighting towers if the match takes place at night. To identify the greening range of the playground, initially, some frames of the empty playfield are taken using background scanner bee. The scanner bee moves diagonally from the corner to the playground center points taking a set of consecutive frames. 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. At the end of this phase, a range of the hue component of the playfield greening color are formed.

On the other hand, the main target of the object detection phase, which is activated as long as the match continues, is to detect the ball in the current frame. It consists of four sub-phases namely; Background Subtraction, Edge detection, Ball Candidates Identification, and Classification. During the first sub-phase, for the current frame, the RGB color space will be converted to HSV one, and 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. And then the matching values will be removed from the current frame to exclude the playfield background color. During the edge detection sub-phase, the canny edge detector is applied to detect the frame objects. It works by detecting discontinuities in brightness, hence, finding the boundaries of objects within the image. During the third sub-phase, a set of candidate balls are identified from the objects founded in the frame using the Circle Hough Transform technique. Finally, during the last sub-phase, a new fuzzy based inference engine is used to confirm the real ball. Three different fuzzy sets are considered, which are; Form Factor FF, Area Ratio AR, and Eccentricity EC. Employing fuzzy logic has a significant impact on promoting the performance as it maximizes the ball identification. The proposed FB²D has been compared against recent soccer ball detection strategies, experimental results demonstrate that FB²D 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 of this paper is organized as follows: section 2 introduces the background and basic concepts, section 3 introduces the previous efforts in the area of ball detection, section 4 presents the problem statement, section 5 presents the proposed system, section 6 presents the performance analysis and experimental results, section 7 introduces our proposed system pros and cons, section 8 summarizes our conclusions.

2. Background and Basic Concepts

In this section, a brief explanation of camera setup for ball detection and tracking will be introduced. And also, the general procedure for the soccer ball detection will be introduced.

2.1. Camera Setup for Ball Detection and Tracking

The camera setup is a crucial part of any ball-detection system. A high image quality camera with a clear field of view contributes to a smooth detection process. In this regard, the placement of the cameras has to be in such a way that covers the whole playing field and does not interfere with the game in any way. Many times, along with the ball, other objects of interest such as the players and the court lines also need to be detected along with the ball, and the camera setup has to take this into account as well.

As illustrated in Figure 1, four different approaches can be used for ball detection and tracking; they are: (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 an easy use. It is employed in [22], where a single wide-angle fish-eye lens camera for a Raspberry Pi is used to record the videos. Also, the work introduced in [23] used a single-camera view from behind the tennis-court. On the other hand, MCA, is generally used to handle occlusion efficiently or for mapping larger areas. Although MCA provides a significant advantage in localizing the position of the ball, its setup is complex in comparison to SCA. In [24], multiple high-performance cameras set up in the tennis court for broadcasting purposes are used. In [25], nine IP cameras as well as one overhead camera were utilized. This setup covers the entire court region from every corner of the court. The system introduced in [26] utilized a dual-camera setup with two smartphones placed behind the court along the sidelines to record their videos. Using accurate cameras of the smartphones is a convenient approach as they give a high-quality imagery for ball detection.

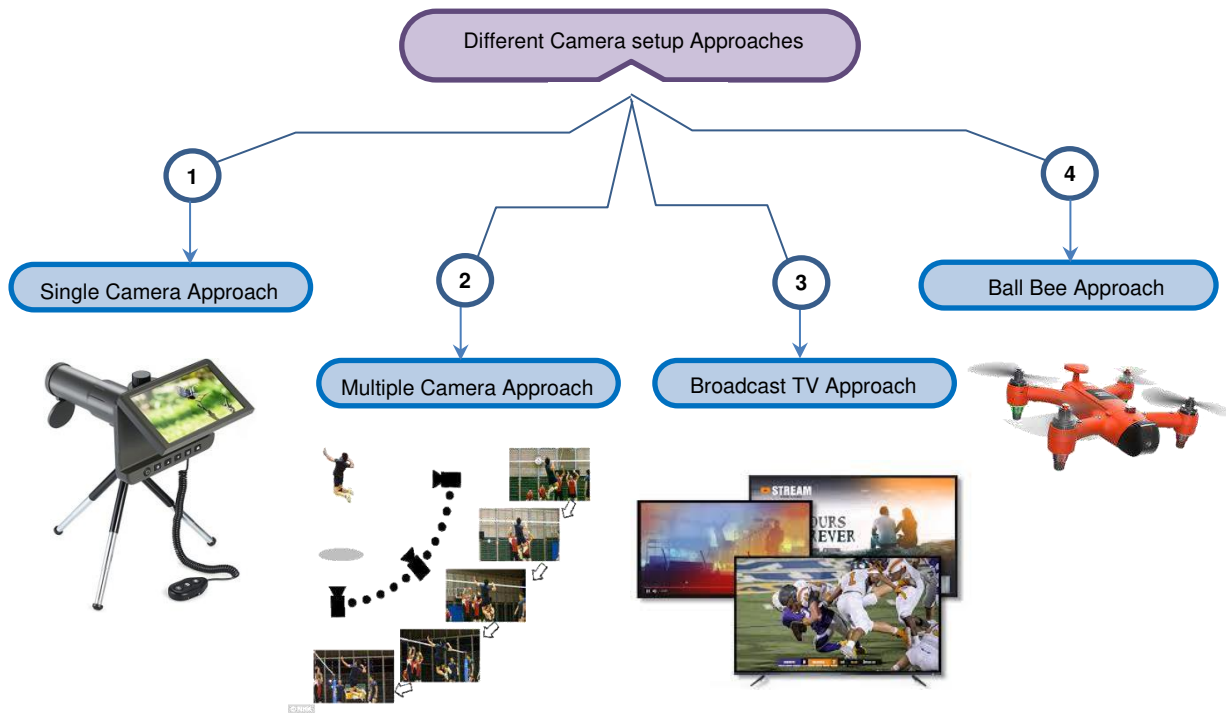


Figure 1 Different camera setup for ball detection

Another way for obtaining a dataset for both ball detection and tracking is BTVA, which utilizes the broadcast TV videos of the sport. Although BTVA makes it easier for obtaining the game videos, it suffers from various camera angles and pan, tilt, zoom, and motion of the camera in a broadcast video [27] [28] [29]. 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. In [38], a method for tracking the soccer ball via a quadcopter is presented; such quadcopter is called the ball bee. The main components of this bee are the four motors and the front camera. The proposed tracking technique is divided into two different modes: the first mode is the Previous Frames Dependent (PFD) mode. In this mode, the bee moves according to the position of the ball in the previous and current frames. The direction of the vector between the two positions will be the direction of the movement. In the second mode, called Frame Center Dependent (FCD) mode,

the direction of the vector between the center of the frame and the position of the ball in the current frame will be the movement direction of the bee.

2.2. General Procedure for Soccer Ball Detection

As shown in Figure 2, ball detection can be accomplished 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, which are background modeling and ball modeling. For background modeling, a set of input frames (initial frames) are employed to model the dominant background, while the employed ball (may be a set of balls) is also considered to build the ball model that will be employed through the next phases to identify the candidate balls. The generated ball model summarizes the ball features such as size, color, shape, etc. 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. 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. In order 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 PP phase through the ball modeling process to identify the correct ball if exists. This can be accomplished during BDP through ball validation process. Finally, the ball can be localized within the frame to update the template and to guide later possible ball re-detection.

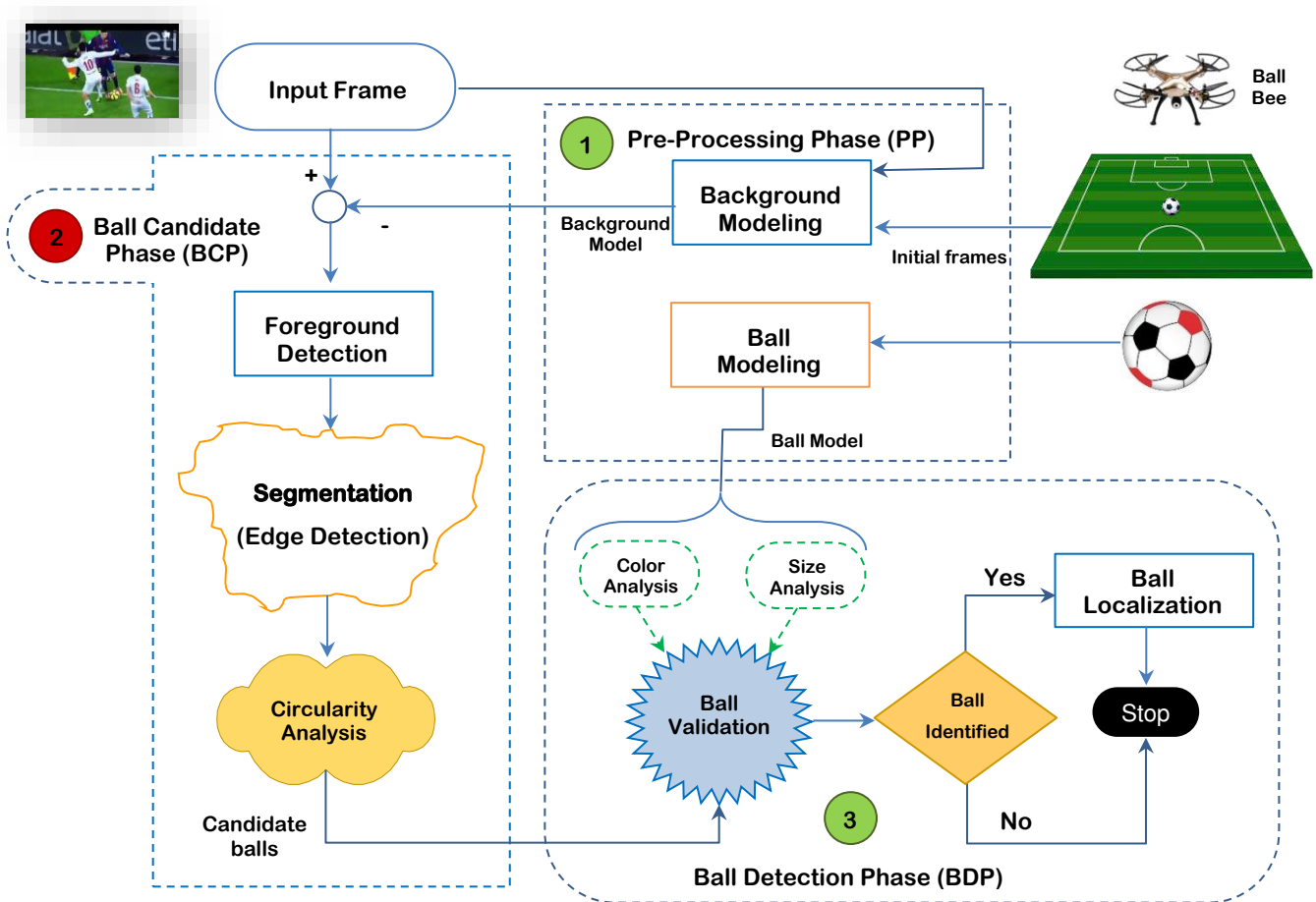


Figure 2 General Procedure for Soccer Ball Detection

3. Related Work

In soccer videos, the most significant object is the ball where its detection is motivated by various applications (e.g., event detection, tactics analysis, automatic summarization, and object-based compression) [30]. Methods of ball determination in soccer videos can be divided into two groups: the first one uses fixed cameras - usually calibrated in advance - in a controlled environment [31]; the second group just employs the regular broadcasting videos. Whereas the first group can provide better performance, the latter one is considered more flexible. The ball is detected by both its chromatic and morphological features. It is supposed to be among the white regions and its circularity that gives an indicator of resemblance of the shape to a circle is bigger than the given threshold. In [32], a modified Circle Hough Transform is proposed to detect the ball on selected frames. This method, however, cannot handle occlusions and it requires the ball to be homogeneous, which greatly limits its application. In [33], there is an application of ball detection for video compression. Color histogram back projection and intensity template matching are employed to detect the ball. It is claimed that trajectory knowledge is used for prediction, but no details had been given in that paper.

[34] illustrated Hough transform voting scheme. Unlike the object-based algorithms, a trajectory-based algorithm was put forward in [35] for both ball detection and tracking. In this off-line framework, how to obtain the ball candidates removes the false alarms and verifies the candidates that become critical. Similarly, trajectory constraints are also used in [36], where the difference is that a Viterbi algorithm rather than a Kalman filter is used to verify trajectory candidates of the ball. As soccer is a sport that has spectators; the playfields, the lines, the ball and the clothing of the players are designed to be visually distinctive in color. Therefore, some approaches sought to alleviate detection difficulties by finding the grass playfield first [37], using color segmentation and post-processing with morphological operations like connected component analysis to limit the search area. However, how to remove false alarms (e.g., small areas that look like the ball) and cutoff artifacts, especially when player blobs merge with the field lines, is still a challenging problem. A simple way to represent the playfield color is using a constant mean color value that is obtained through prior statistics over a large data set. The color distance between a pixel and the mean value of the field is used to determine whether it belongs to the field or not [38].

As the color of soccer field is a rich green one, the hue component mentioned in Smith's hexagonal cone model can be employed for detecting the green colored pixels within a certain range [39]. In [40], R-D method is proposed and the authors used the image distinctions and time variance to calculate the foreground pixels; although this method is suitable for real-time application, it produces insufficient accuracy in case there are multiple objects in a scene. A statistical method is proposed in [41]; it does not update the background pixels that makes it less useful under a changing illumination. In [42], a recursive filter is used for updating the background that solely depends on the learning parameters. The dependency on the learning parameters may cause either trail or delay in updating the background model. In order to extract the object, a frame difference method is proposed in [43]. However, the frame difference method stops the extraction procedure when the target becomes stationary. In [44], Yao and Ling proposed an improved version of GMM, but this version failed in detecting the object near camouflage region. Previous studies reveal that the sample size of object is either lost or buried under not only the noise but also complex condition. Even though some methods are applicable in real-time scenario, most of them suffer from either ghost effect or aperture distortion.

Methods depending on fast learning parameter cause trails behind the object and reduce the accuracy, while other methods having low learning rate do not accurately update the background. Therefore, a regional level processing may be beneficial to update only the changing background pixels and to detect the actual non stationary pixels. In [45], they integrated the regional level of processing by evaluating the block wise entropy and energy, which provided the actual moving pixels on the foreground. The good object detection results are obtained from the background subtraction method, which depends on reliable background building techniques; many researches have proposed various methods to create background models. For example, [46] proposed an approach estimating the background intensity variation of an input image and subtracting this estimation from the input to achieve a flat-background image, which is processed by a global thresholding approach to get the final binary outcome. However, this method for estimating the

background through the iterative operations from a single image has a high computational cost. It is not suitable for application to real-time systems, though. [47] used the color change of continuous images to determine the background-image convergence time, and a two-layered image pyramid structure is proposed.

Based on the image pyramid concept, they calculated the stable sum of absolute differences (SAD) value for low-dimensional images and adjusted the weight of the color difference between the foreground and background images according to this value to reduce interference from moving objects or noise. When the color change of an image tended to be stable, the image was converged into the background. Furthermore, this method cannot create a background when the background is unstable. Statistical modeling was also used to detect moving objects in images. This method approximates the probability of density functions of real images by using specific probability density functions in the feature space, and it estimates the occurrence probabilities of subsequent images with a high occurrence probability are regarded as backgrounds. Clustering has also been used to establish backgrounds. For example, [48] used a fixed number of categories to cluster input images over a period of time and treated the clustered category as a background category. In case the difference between the subsequent input image and the background category is too large, the category will be treated as the foreground. [49] used a color classifier to create color categories and proposed the concept of category probability.

Table 1 Previous Work Researches

Year	Paper	Contribution	Advantages	Disadvantages
2020	[54]	Proposed a Metaheuristic algorithm for the detection of the sports objects from the video sequence.	<ul style="list-style-type: none"> rectified the challenges in object detection due to shadows, lighting, and detected the fast-moving player while hitting the ball etc. 	<ul style="list-style-type: none"> Occlusion problem Efficiency Problem
2018	[46]	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.	<ul style="list-style-type: none"> 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	[51]	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.	<ul style="list-style-type: none"> 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	[56]	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.
2010	[49]	Used a color classifier to create color categories and proposed the concept of category probability.	Consider shadow removal and object rebuilding for enhancing the foreground objects.	Required a large amount of memory and long processing times.

An efficient ball recognition system working on real images with variable light conditions and non-controlled backgrounds has been implemented in [50]. The choice of the method has mainly been done taking into consideration both computational aspects (the soccer context requires real time processing) and system performances (false positive are not allowed if the goal event has to be detected). While in [51], Fast Robustness Feature detection of SURF algorithm is employed to each frame image for a quick locating of an object and displaying its information. Affine transform is used to estimate transformation. The experimental results showed the capability and robustness of recognizing objects. In [52], an initial background extraction algorithm called Entropy-Based initial Background Extraction (EBBE) was proposed for poor initial-background conditions. The EBBE algorithm reduces the generation of false foregrounds in circumstances where there are large color disturbances due to complex conditions (e.g., camera shake, shaking trees, and snowflakes, or due to a very low frame rate). In [53], a Unified model (Yolov3-Improved Non-Maximum Suppression (INMS)) method is proposed to increase the performance in moving object detection. The experimental result showed that the developed Unified model method has the highest performance in the object detection in comparison to the other methods. A Metaheuristic algorithm for the detection of sports objects from the video sequence is proposed in [54]; such algorithm also rectified the challenges in object detection due to shadows and lighting. The results showed that Cuckoo Search Algorithm could give better accuracy and precision with respect to recall than the other algorithms.

There are numerous automatic foreground extraction methods for a one frame processing [55-58] or sequence frames processing [59-62]. [55] employed a histogram learning technique to detect the background pixels. Color models determined the foreground and background pixels via a training set of soccer videos. The color model is an RGB color histogram with N bins per channel in the RGB color space. [56] employed their algorithm for a soccer field and assumed that the background color is a green color. They will determine the background pixel in case the color intensity is $G > R > B$. [57], [58] extracted the foreground from the background by using Euclidean distance function of the determined pixels as a foreground and surrounding pixels. If the distance is less than a threshold, the pixel will be a foreground. They calculated the distance in RGB color space [57] and HSV color space [58].

[59] used a background subtraction method along with a frame differencing method via three consecutive frames, which were employed for segmentation on the moving objects in the foreground. [60] accomplished the segmentation of background by using the dominant color distribution. They calculated the mean of the peak intensity values of the histogram of several frames. And then, they measured each pixel intensity respecting to the mean value by using Mahalanobis distance. In case the distance is less than the variance, the pixel will be supposed to be background. [61] extracted the interest object in dynamic background via velocity histogram based on Harris corner detector and Lucas-Kanade tracker. Moreover, [62] extracted the interested objects as [61] and shifted the histogram based on motion history image (MHI). [63] aims at building an algorithm for background detection based on the dominant color in RGB color space. This algorithm has robust and fast processing time to be used in robot application, such as a soccer robot.

As depicted in table 1, traditional ball detection methods deal well with situations where ball is visible as a single object, separated from the player body. They have problems to detect the ball when it's possessed or partially occluded by a player.

4. Problem Statement

Motion detection is the simplest one of the three-motion tasks related to detection, segmentation and estimation. Its goal is to identify which images' points and more generally which regions of the image have moved between the two-time instants. The motion of image points is not directly perceived but rather through intensity changes. However, such intensity changes over time may also be induced by camera noise or illumination changes. In addition, the object motion itself may induce small intensity variations or even none at all. Moving objects often convey the essential meaningful information in videos. In videos, automatic detection of moving objects is the first crucial stage for various potential applications (e.g., pedestrian and vehicle tracking, action and event recognition, annotation of video archives, and etc.). There is no doubt that detection of moving objects at any scene is an important topic in the field of computer

vision. It is one of the basic steps in many vision-based systems. For instance, applications such as human computer interface (HCI), robot visions, and intelligent surveillance systems require detection of moving objects. Various methods have been proposed and they proved a successful method for detection of moving objects in case of stationary cameras. On the other hand, in case of using mobile or pan-tilt-zoom (PTZ) cameras, these methods do not work well due to many unaccounted factors that arise while using of movable cameras.

An ability of both accurate ball detection and tracking in a video sequence is a core capability of any system, which aims at an automatic analysis of football matches or players' progress. Unfortunately, ball detection through long-shot video footage of a football game is not a trivial task for the following reasons: (i) The object of interest (e.g., the ball) has very small size compared to other visible objects in the observed scene. (ii) Ball size varies depending on its position on the play field due to the perspective projection. (iii) The ball shape is not always circular. (iv) When a ball is kicked, it moves at a high velocity, accordingly, its image becomes blurry and elliptical. (v) Perceived color of the ball changes due to shadows and lighting variation. (vi) The ball color is similar to the color of white lines on the pitch and sometimes to players' jerseys. (vii) Other objects with similar appearance to the ball can be visible, such as small regions near the pitch lines and regions of players' bodies such as a head. (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.



Figure 3 A High Variance in Ball Appearance

5. The Proposed Fuzzy Based Ball Detection (FB²D) strategy

From a simple point of view, ball detection can be considered as the problem of detecting circular features in sequences of images when the light conditions produce the ball appearance in the image as a circular pattern. In fact, when using artificial lights (matches played in the evening) and shaded natural lights, the ball results are very close to a spherical object in the scene. Unfortunately, in a directional and dominant sun light the ball appears in the image as a spherical cap whose orientation varies according to the sun orientation in the stadium: the detection and recognition techniques have to be tuned to detect different patterns the ball maps into the image due to the particular case, in which we have to deal with.

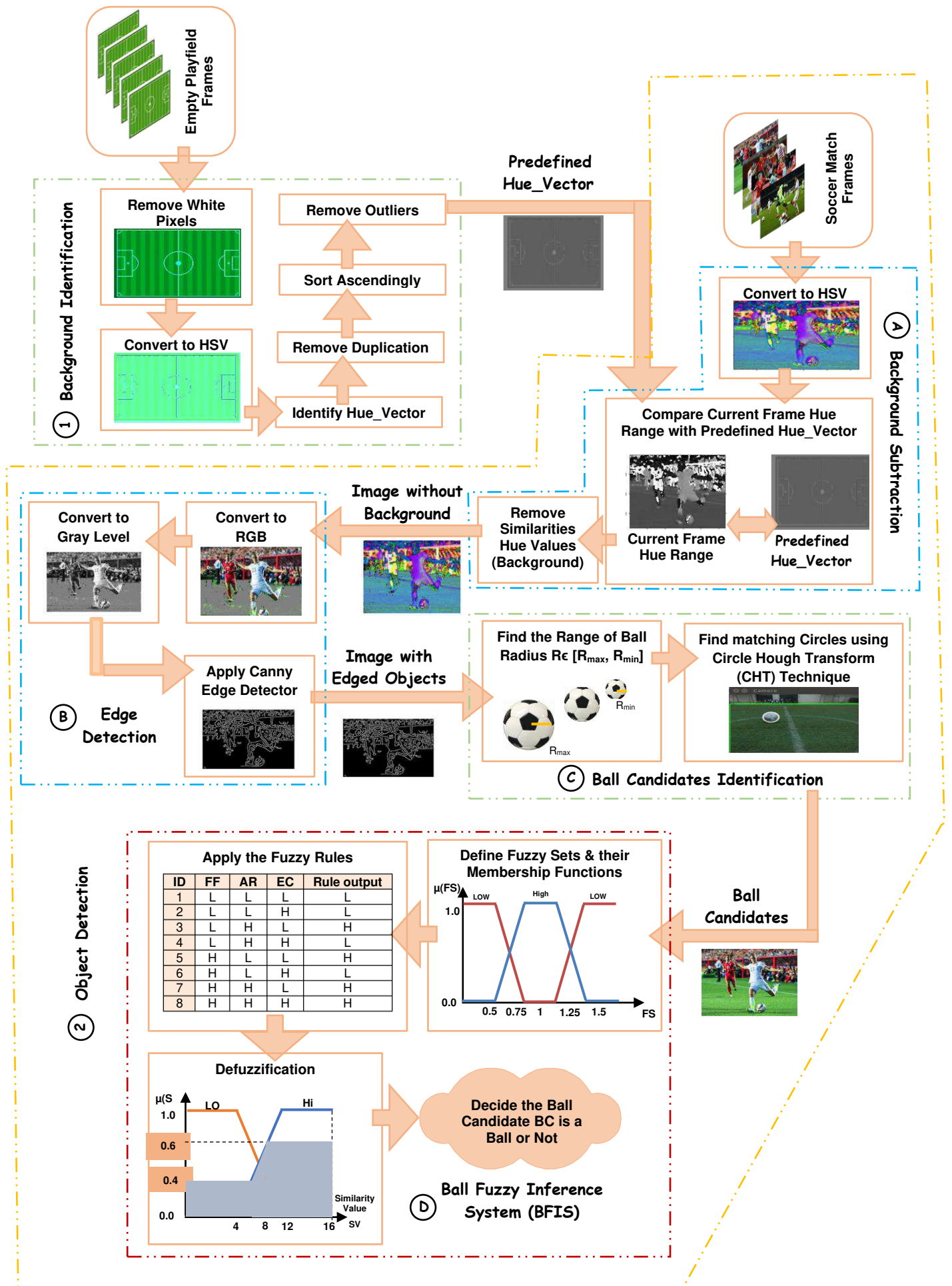


Figure 4 Fuzzy Based Ball Detection (FB2D) strategy Framework

There are various methods for the detection of a moving object such as: (i) optical flow method, (ii) consecutive frame difference, and (iii) background subtraction. In Optical Flow Method, an automatic feature extraction has been done via clustering; consequently, features are extracted from the current image by using x-mean cluster and the classification of these extracted features will depend on their estimated motion parameters. The segmented region is labeled and labeling result is characterized as a moving object. Furthermore, it cannot be employed to the real time application without using some special hardware. Consecutive frame subtraction is an easy method and it also works well in dynamic environment although it does not completely extract the moving object. Gaussian mixture model and water-shed are used in this method. Firstly, it is proposed that where the distinction between the two frames is calculated, and then dividing it to both a moving area and a background one. Background subtraction is a method in which incoming frames are compared to the background model, and then the moving object will be detected. There is also a difficulty in such situation where background is kept on changing.

Fuzzy Based Ball Detection (FB2D) strategy

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 [R_{min} , R_{max}]

Output:

The object is a Ball or not

Steps:

// **Background Identification**

1. **For each** frame fe_i **in** frame set **FE** **Do**
2. Remove white pixels
3. Convert to HSV
4. Identify Range_H(fe_i)
5. Add Range_H(fe_i) to H_Vector
6. **End For**
7. Remove duplication of H_Vector
8. Sort ascendingly of H_Vector
9. Remove outliers from H_Vector

// **Object Detection**

10. **For each** frame f_i **in** frame set **F** **Do**
11. Convert to HSV
12. Compare Range_H(f_i) to H_Vector
13. Remove similarities
14. Convert to RGB
15. Convert to Gray level
16. Apply Canny Edge Detector
17. Use Circle Hough Transform (CHT) with radius range [R_{min} , R_{max}]
18. **For each** object ob_i **in** frame f_i **Do**
19. Define fuzzy sets (FF, AR, EC)
20. Find the input membership functions
21. Apply fuzzy rules to the inputs
22. Defuzzify the output of the fuzzy rules using output membership function
23. Use the center of gravity method:

$$COG = \sum \mu(SV) * SV / \mu(SV)$$

24. **If** COG >= β **Then**
25. Object ob_i is the ball
26. Goto aa
27. **Else**
28. Object ob_i is not a ball
29. Goto bb
30. **End IF**
31. **bb: End For**
- aa: 32. **End For**

Algorithm Parameters	
FE	The frame set of empty playfield before the match starts.
F	The frame set of playfield when the match starts.
R_{min}	The minimum radius of the ball in the frame
R_{max}	The maximum radius of the ball in the frame
Range_H(f_i)	The range of hue values in frame i
H_Vector	The range of hue for the background or the playfield
FF	The Form Factor feature of object
AR	The Area Ratio feature of object
EC	The Eccentricity feature of object
SV	Similarity Value
$\mu(SV)$	Membership function of Similarity Value
COG	The Center of Gravity method to calculate the bounded area under a curve

Algorithm 1 Fuzzy Based Ball Detection (FB2D) strategy

One of the methods widely used for detecting a moving object is the background subtraction method. The major reason for using this method is its simplicity; it is accurate and takes less computational time. Like other methods background subtraction method also has to face some challenges like system limitations and environmental changes. System limitation means the platform, on which application has been used while the environmental changes mean variations in illuminations, lights, shadows, colors similarity, etc. The proposed system in this paper is called Fuzzy Based Ball Detection (FB²D) strategy. This approach 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

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 pixels that form the playfield. The soccer playfield has a one distinct, dominant color (or a tone of green color) that may vary from a stadium to another in addition to the weather and lighting conditions within the same stadium. Thus, we do not assume any specific value for the field color in the suggested algorithm. This section delineates a method for detecting values of the playfield color. As the standard RGB color representation is not suitable, the RGB values of the captured frames are transformed into corresponding coefficients in the HSV color space [64] before the analysis process.

Furthermore, HSV represents three diverse parameters and they are as follows: hue, saturation, and brightness. Hue determines the dominant wavelength of the color with values ranging from 0 to 360 degrees. Brightness illustrates the level of white light (0 - 100) while saturation describes the proportion of chromatic element in a color. Because humans are much more sensitive to hue rather than saturation and brightness, one parameter becomes far more important than the others and the HSV representation, therefore, is excellent for color analysis.



Figure 5 The Soccer Playfield

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 four 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 $F \in \{f_1, f_n\}$ 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 in the beginning of the match 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 than the threshold value as in (1), these pixels will be removed from the frame.

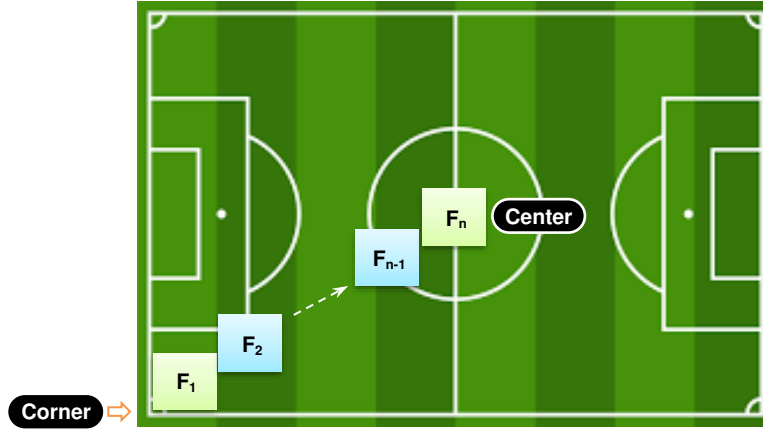


Figure 6 The Frames along the diagonal from one corner to the center of the playfield

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

Where P is any pixel in the frame f , (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 β 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 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, and etc.). Although different color variation can be seen in these areas, the hue for all these regions do not vary much, so hue value can be useful in soccer playfield identification. The formulae for converting RGB to HSV color space can be referred to [65]. Fourthly, for each frame f_i in the frame set F , the value of the hue component for each pixel P will be determined to identify the range of hues, which will be called $\text{Range}_H(f)$. 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 F . And then, the duplicated values will be removed from this vector and it will be sorted ascendingly. Finally, the extreme values, in this step, will be considered as outliers and they have to be removed from the H_Vector .

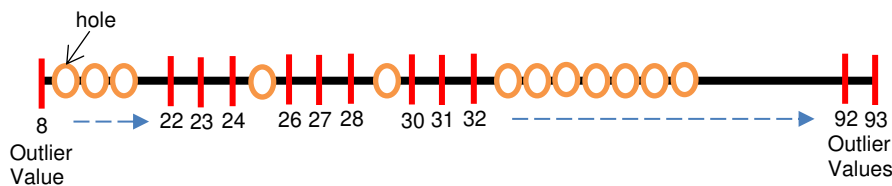


Figure 7 The Range_H (Green) - for example

For instance, as the values of the H_vector shown in Figure 7, there are some holes in different places. In Figure 7, the number line starts at 8, and then there are some holes until numbers from 22 to 24 appear. After that, there is only one hole and then numbers from 26 to 28 appear. Again, there is one hole and then the numbers from 30 to 32 appear; later, there are some holes until the last range of numbers from 92 to 95 appears. Moreover, the holes in this case will be defined as the lost adjacent values in the H_Vector . In case there are few holes, their values will be considered in the range of hues (H_Vector), but if there is a plenty of holes between one sequence of numbers and the next sequence of numbers, the last sequence of numbers will be considered as an outlier and it will be removed from the H_Vector . These outliers can come from

any thing will be seen in the playground. In Figure 8, the playfield looks like an empty field, but actually 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.

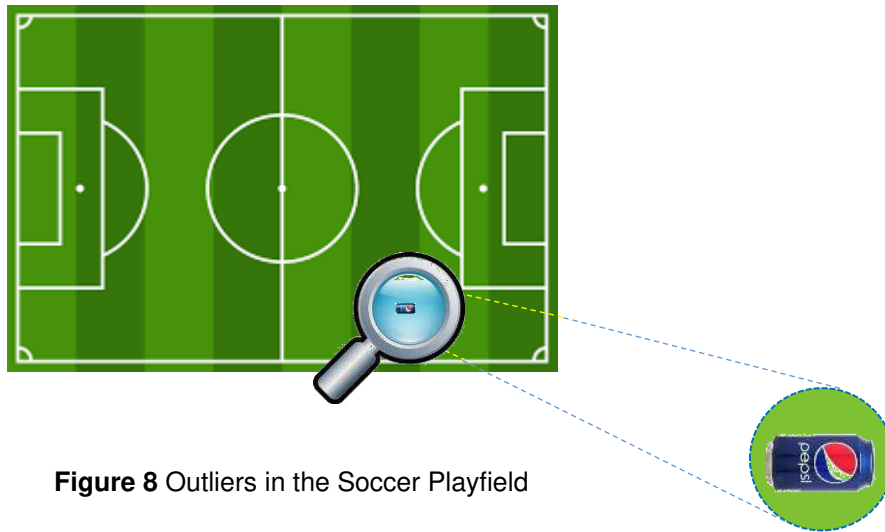


Figure 8 Outliers in the Soccer Playfield

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 Ball Classifier. These steps will be explained in details in the next sections.

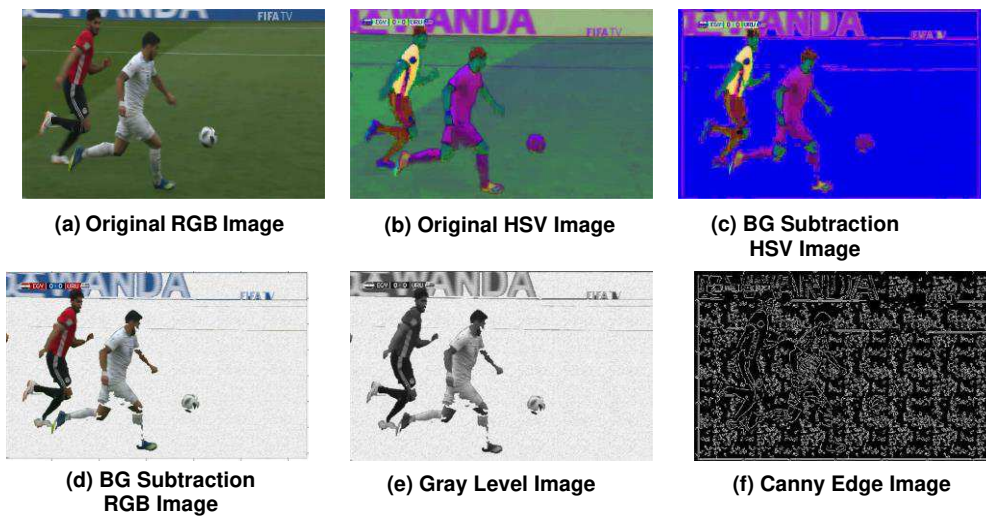


Figure 9 Different States of the Image

5.2.1. Background Subtraction

Background subtraction is a method which is basically used for detecting a foreground object from an image through comparing two different frames. Basically, it will compare the value of the difference to the threshold value. The threshold value is not predefined, but it will be calculated via the first few frames that are given. Object detection is performed to check the existence of an object in a video as well as locating it. As the challenges that have to be faced during background subtraction are frequent changes due to illumination changes, motion changes, and background geometry changes; there are many different background subtraction methods like frame difference, Gaussian mixture model, kernel density estimation and codebook.

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, the hue values will be compared to the predefined hue values in the H-Vector from the previous section. Finally, in the fourth step, the pixels with the same predefined hue values as in the H_Vector will be defined as the background pixels in the frame. Hence, these background pixels will be removed from the frame. As shown in Figure 9(d), the remained pixels in the frame will be the moving objects, which are the players and the ball.

5.2.2. Edge Detection

Edge detection refers to the process of both identifying and locating sharp discontinuities in an image; these discontinuities are abrupt changes in pixel intensity scene. It works by detecting discontinuities in brightness and is used for both image segmentation and data extraction in areas such as image processing, computer vision, and machine vision. Edge detection is an important technique in many image processing applications (e.g., object recognition, motion analysis, pattern recognition, medical image processing, and etc.). The basic idea behind edge detection is to find places in an image where the intensity changes rapidly. There are two general standards as mentioned in [66]. In the first technique; the places where the first derivative of the intensity is greater in its magnitude than a specified threshold has to be located. Otherwise, in the second one, the places where the second derivative of the intensity has a zero crossing has to be located. Generally; the syntax for edge function is as mentioned in (2).

$$[g, t] = \text{edge}(img, 'method', parameters) \quad (2)$$

Where *img* is the input image, *method* is one of edge detection techniques and *parameters* are additional ones. Edges in an image are pixel locations with abrupt changes in gray levels. In our paper, the canny edge detection technique [67] is used for detecting the object. It is a method to find edges by isolating noise from the image without affecting the features of the edges in the image and then applying the tendency to find not only the edges but also the critical value for threshold. In this section, the canny edge detector will be used for defining the edges of the moving objects as illustrated in Figure 9 (f).

The algorithm of the Canny edge detector has three phases. Firstly; the noise of the image has been filtered out by using a Gaussian smoothing filter. Secondly; the edge strength in the smoothed image is located by computing the image gradient, which helps to indicate where the actual edge is located. Thirdly, the last step is to thin down the edges by tracking along the edge in the edge direction and set any pixel that is not at the maximum to be 0, which is called non-maximum suppression. Eventually, edges are detected and linked using pixel connectivity and double threshold. This means that, if the magnitude is above the high threshold, it is considered as an edge. In case the magnitude is below the low threshold, it is considered as a non-edge. This edge detector has the advantage that maximum edges get detected via this edge detector. The canny edge detector is able to detect maximum number of edges. Canny's edge detector gives very good results for detecting horizontal and vertical edges. It can also detect edges at the corner and the circular edges which are suitable for soccer ball detection.

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 are the ball, the referee, the players and some noises. The Circle Hough Transform (CHT) [68] aims at finding circular patterns of the given radius R within an image. Each edge point forms a circle of radius R to an output accumulator space. The peak in the output accumulator space is detected where these contributed circles will be overlapped at the center of the original circle. In order to reduce the computational burden and the number of false positives typical of the CHT, a number of modifications has widely been implemented in the last decade. The CHT can be formulated as convolutions applied to an edge magnitude image (after a suitable edge detection). In order to use the Circle Hough Transform (CHT) technique, the range of the expected ball radius values need to be identified as shown in Figure 10. Consequently, before the soccer ball match starts, the quadcopter camera 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_{max} of the ball. On the other frame, the ball will be in the corner of the frame to find the minimum radius R_{min} . From these two positions of the ball in the frame, the range of ball radius values will be determined as $(1 - \rho) * R_{min} \leq R \leq (1 - \rho) * R_{max}$, where $0 \leq \rho \leq 1$.

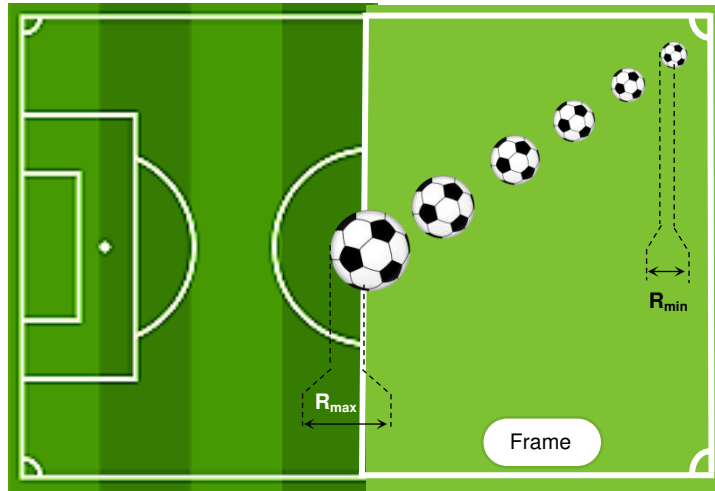


Figure 10 Identify the Range of Ball Radius Values

5.2.4. Ball Fuzzy Inference System (BFIS)

After ball candidates have been identified over the current frame, a Fuzzy Inference System is used to decide whether the candidate is a ball or not. In this system, 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.

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 (3).

$$FF = 4\pi A / P^2 \quad (3)$$

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 11 (a), 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;

$$FF = 4 * \pi * \pi * r^2 / (2 * \pi * r)^2 = 4 * \pi^2 * r^2 / 4 * \pi^2 * r^2 = 1.$$

On the other hand, if the object is a square with length L as in Figure 11 (b), the Form Factor will be;

$$FF = 4 * \pi * L^2 / (4 * L)^2 = 4 * \pi * L^2 / 16 * L^2 = \pi / 4 = 0.785.$$

The second feature needed to be evaluated is the Eccentricity, which is calculated as shown in (4).

$$EC = c/a \quad (4)$$

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 11 (a), 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 (5).

$$AR = \frac{A_{Obj}}{A_{MBR}} \quad (5)$$

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 11 (a), 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 $2 * r$; the area of the bounding square will be $(2 * r)^2 = 4 * r^2$. The Area Ratio of the circle will be $AR = \pi * r^2 / 4 * r^2 = \pi / 4 = 0.785$. Consequently, the optimal Area Ratio of a circle is 0.8.

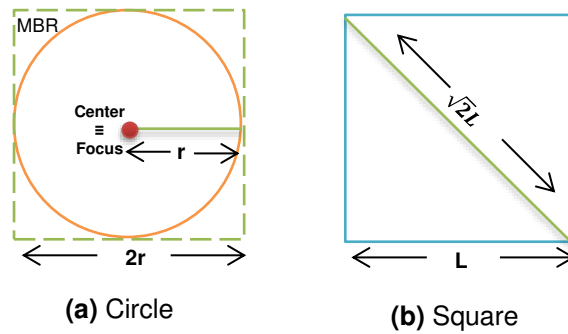


Figure 11 Illustrated Example Shapes

As illustrated in the previous paragraphs, a candidate ball may be 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. Fuzzy Inference System is suitable for approximate reasoning as it can be used efficiently in decision making under incomplete or uncertain data. Hence, Fuzzy Inference System can be successfully employed to assign a similarity value for each ball candidates. Generally, Fuzzy Inference System can be applied through three sequential steps, and they are: (i) fuzzification of inputs, (ii) Fuzzy Rule Induction, and (iii) defuzzification.

a. Fuzzification of Inputs

Three different fuzzy sets, which are FF (BC), AR (BC), and EC (BC) will be considered. During fuzzification, the input crisp values are mapped into grades of membership for linguistic terms, “Low” and “High” of the used fuzzy sets. This can be accomplished by employing a membership function for each fuzzy set, which provides the similarity degree of a crisp input to the corresponding fuzzy set. Such process returns a value between 0.0 and 1.0 for non-membership and full-membership, respectively. The employed membership functions for the considered fuzzy sets (e.g., FF , AR , and EC) are illustrated in Figure 12.

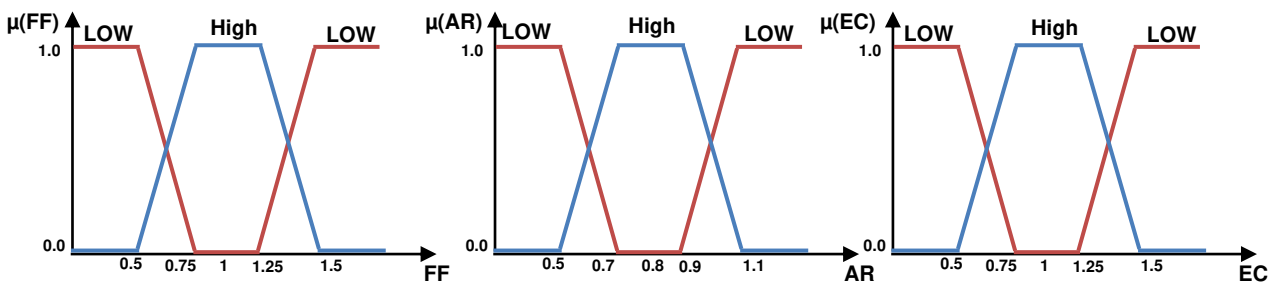


Figure 12 Input Membership Functions

b. *Fuzzy Rule Induction*

For the inference process, a set of fuzzy rules are employed in the form *if (A is X) AND (B is Y) AND (C is Z) THEN (D is M)*, where *A, B, and C* represent the input variables (e.g., *FF, AR, and EC*), while *X, Y, and Z* represent the corresponding linguistic terms (e.g., *low or high*), *D* represents the rule output, and finally *M* represents a linguistic term (low or high). Hence, the output of the fuzzification is the input for the fuzzy rule induction. There are 8 rules that are listed in Table 2 (assuming ‘*L*’ refers to “Low”, ‘*H*’ refers to “High”). For more illustration, the first rule in Table 2 indicates that *IF FF is Low AND AR is Low AND EC is Low THEN Output is Low*.

Table 2 The used fuzzy Rules

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

Generally, there are four fuzzy rules inference methods, namely; max-min, max-product, sum-dot, and drastic product. The max-min is the used method in this paper. It is based on choosing the min operator for the conjunction in the premise of the rule and for the implication task, while the max operator is used for aggregation. Hence, for *w* input variables and *q* states of the output linguistic terms, the max-min inference rule can be illustrated in (6).

$$\mu_{out}(x) = \overbrace{MAX}^{aggregation} \left[\underbrace{MIN}_{implication} (\mu_{inp(1)}, \mu_{inp(w)}) \right] \forall x \in \{1, 2, 3, \dots, q\} \quad (6)$$

c. *Defuzzification*

The output of the Fuzzy Inference System is a fuzzy set. However, crisp values are required for most real-life applications; thus, the output of the fuzzy rules should be defuzzified. The most common defuzzification technique is the Center of Gravity (COG). Based on COG, the weighted average of the area bounded by the membership function curve is computed as in (7) to be the crisp value of the fuzzy quantity. In our case, defuzzification was accomplished 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(BC)*; the output value of the defuzzification process is a crisp value that expresses the Similarity Value (SV) of the ball candidate *BC*, e.g., *SV(BC)*. Finally, the decision is taken whether *BC* is a ball or not based on a simple rule, which is expressed by the simple step identification function illustrated in Figure 14.

$$COG = \sum \mu(SV) * SV / \mu(SV) \quad (7)$$

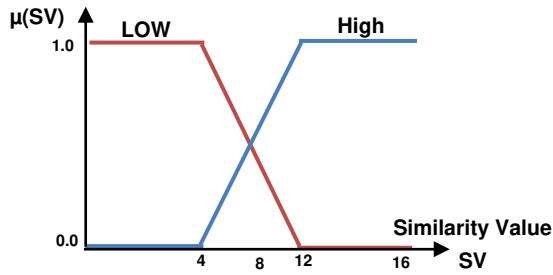


Figure 13 The Output Membership Functions

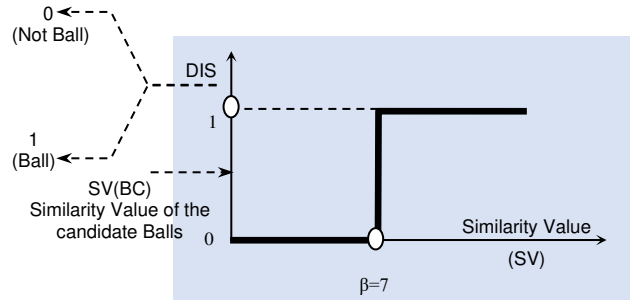
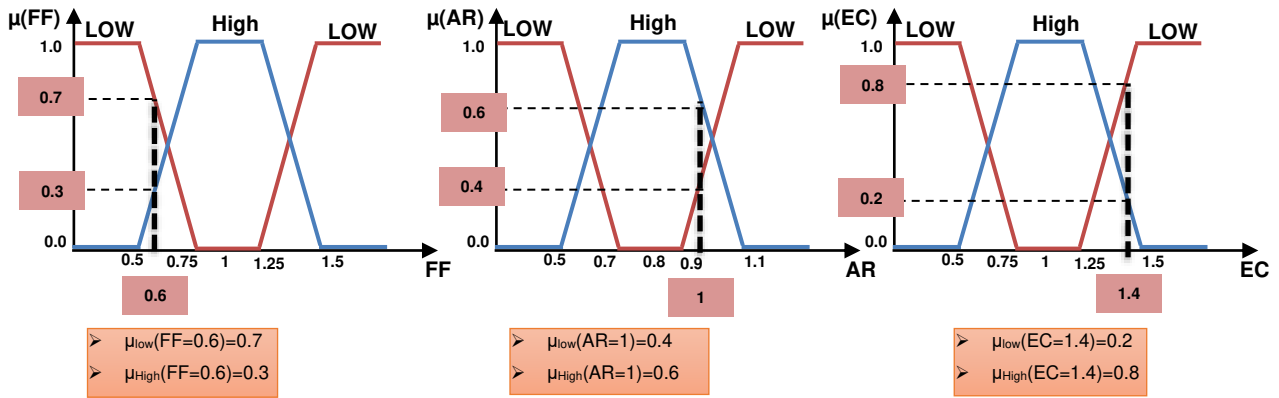


Figure 14 The used identification function

Step (1) Find the fuzzy set membership values for the ball candidate BC (FF=0.6, AR=1, EC=1.4)

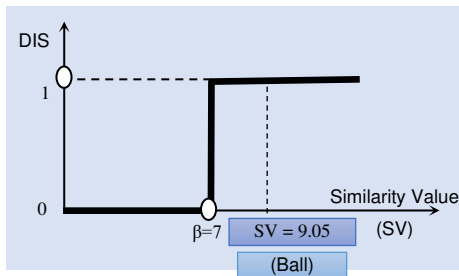


Step (2) Calculate the fuzzy outputs for each rule.

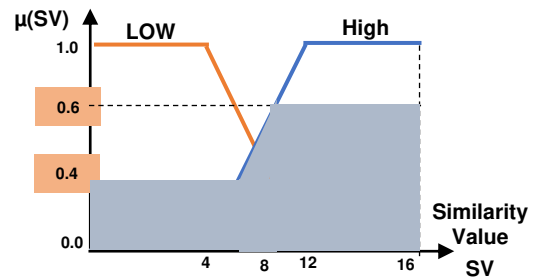
ID	FF	AR	EC	Rule output	Output Membership
1	L	L	L	L	MIN (0.7, 0.4, 0.8) = 0.4
2	L	L	H	L	MIN (0.7, 0.4, 0.2) = 0.2
3	L	H	L	H	MIN (0.7, 0.6, 0.8) = 0.6
4	L	H	H	L	MIN (0.7, 0.6, 0.2) = 0.2
5	H	L	L	H	MIN (0.3, 0.4, 0.8) = 0.3
6	H	L	H	L	MIN (0.3, 0.4, 0.2) = 0.2
7	H	H	L	H	MIN (0.3, 0.6, 0.8) = 0.3
8	H	H	H	H	MIN (0.3, 0.6, 0.2) = 0.2

So that the output will be:

- High with membership=MAX(0.6,0.3, 0.3,0.2)=0.6
- Low with membership=MAX(0.4, 0.2, 0.2,0.2)=0.4



Step (3) Defuzzification



Use the center of gravity method:

$$COG = \sum \mu(SV) * SV / \mu(SV)$$

$$COG = \sum_{R=1}^4 R * 0.4 + \sum_{R=5}^{16} R * 0.6 / 4 * (0.4) + 12 * (0.6)$$

Hence the Similarity value SV of BC is 9.05

So that the candidate ball BC is a ball

Figure 15 Illustrative Example to calculate the Similarity Value of a Candidate Ball using a Fuzzy Classifier.

5.3. Ball Detection Optimization: Illustrative Example

In this section, an example about ball candidate classification will be illustrated and explained in details in Figure 15. For example, a ball candidate BC_i 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 = $Output_H = \text{MAX}(0.6, 0.3, 0.3, 0.2) = 0.6$ and Low with membership = $Output_L = \text{MAX}(0.4, 0.2, 0.2, 0.2) = 0.4$. 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 Center of Gravity method as in (7). This result crisp value will identify the Similarity Value of the ball candidate BC_i that will be 9.05. From Figure 14, if the crisp value is greater than β , 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 BC_i will be a ball. $COG = \sum_{R=1}^4 R * 0.4 + \sum_{R=5}^{16} R * 0.6 / 4 * (0.4) + 12 * (0.6) = 9.05$.

6. Performance Analysis and Implementation

In this section, the proposed Fuzzy Based Ball Detection (FB²D) strategy will be evaluated in details. Table 3 illustrates the tunable parameters that will be used in the next experiments with their corresponding values.

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 algorithms are compared with respect to different metrics. The objective is to find out which of the three algorithms can correctly 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 this paper, the different used metrics are precision, recall, accuracy, error and F1 scores as in (8), (9), (10), (11) and (13), respectively.

Table 4 the Classification of the Outcomes of the System

Outcome	Description
A	# of frames where the ball is detected correctly
B	# of frames where the ball is detected incorrectly
C	# of frames where the ball is undetected correctly
D	# of frames where the ball is undetected incorrectly

$$\text{Precision} = \mathbf{P} = \frac{\mathbf{A}}{\mathbf{A+B}} \quad (8)$$

$$\text{Recall} = \mathbf{R} = \frac{\mathbf{A}}{\mathbf{A+D}} \quad (9)$$

$$\text{Accuracy} = \mathbf{Acc} = \frac{\mathbf{A+C}}{\mathbf{A+B+C+D}} \quad (10)$$

$$\text{Error} = \mathbf{E} = \frac{\mathbf{B+D}}{\mathbf{A+B+C+D}} \quad (11)$$

It will be also valuable to combine both precision and recall in one metric. This can be achieved by calculating the F-measure, which can be expressed in (12).

$$\mathbf{F}_n = \frac{(\mathbf{n}^2+1)*\mathbf{P}*\mathbf{R}}{\mathbf{n}^2*\mathbf{P}+\mathbf{R}} \quad (12)$$

Where; \mathbf{n} is a weighting factor, which allows assigning different weights to Precision (P) and Recall (R). The most commonly used F-measure in text classification is F_1 (i.e., $n=1$), which weights equally both precision and recall. It can be expressed in (13).

$$\mathbf{F}_1 = \frac{2*\mathbf{P}*\mathbf{R}}{\mathbf{P}+\mathbf{R}} \quad (13)$$

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 Modified Atherton Algorithm (MAA) [50], Scale-Invariant Feature Transform based Algorithm (SIFT) [21], Speeded Up Robust Features based Algorithm (SURF) [51], Entropy-Based initial Background Extraction (EBBE) Algorithm [52], Improved Non- Maximum Suppression (INMS) [53] and Cuckoo Search Algorithm [54]. Supposing that the camera of the quadcopter is at a fixed height, the ball will correspond to a circular region with radii in the range ($R_{MIN} = 24$, $R_{MAX} = 28$) depending on the distance of the ball with respect to the optical center of the camera. The whole image size was 1280x720 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.

6.3. Evaluating the proposed Fuzzy Based Ball Detection (FB2D) strategy

Our algorithm has been evaluated by comparing it to other different algorithms as illustrated in section 5.2 and Table 5. Our proposed algorithm detects the ball, while the other algorithms may fail to detect it. The experimental results in Figures 18, 19, 20 and 21 show the precision, recall, accuracy, error and F-measure of our algorithm in comparison to the other algorithms, respectively. Our algorithm achieves the highest values for precision, recall, accuracy and F-measure. On the other hand, it achieves the lowest error rate among the other algorithms. Table 6 shows our algorithm results of precision, recall, accuracy, error and F1-measure for all the used ten image sequences.

Table 5 Competitors used for Evaluation

Classification Technique	Description
Modified Atherton Algorithm (MAA) [50]	As stated in [50], the framework consisted of two steps: in the first one, a modified directional circle Hough transform has been used to detect the candidate ball; in the second step, a neural classifier is applied on the candidates to confirm if it is the ball or a not.
Scale-Invariant Feature Transform based Algorithm (SIFT) [21]	It is a method for extracting distinctive invariant features from the images that can be used to perform reliable matching between different views of an object [21]
Speeded Up Robust Features based Algorithm (SURF) [51]	It is a detector of points of interest in an image, wherever the image is remodeled into coordinate's victimization the technique referred to as a multi-resolution. Is to create the copy of the first image with pyramidal Gaussian or Laplacian pyramid form and acquire the image with constant size however with reduced information measure [51]. The standard version of SURF is many times quicker than SIFT.
Entropy-Based initial Background Extraction (EBBE) Algorithm [52]	As in [52], it was used for the detection of 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.
Improved Non-Maximum Suppression (INMS) [53]	INMS is used to detect the multiple objects in less time with higher accuracy. As in [53], 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 [54]	In this algorithm [54], the fitness function evaluation is started by calculating the number of instances in total of each agent α and it also calculates the selection of number of instances in each agent β and the selection of number of boundary instances selected by agent γ . The boundary instances are selected by the algorithm and they are calculated and used for the evaluation of fitness value.

Figure 16 shows the precision for all the ten sequences for the 5th sequence; the precisions are 0.98, 0.95, 0.84, 0.88, 0.95, 0.94 and 0.91 for our FB²S, MAA, SIFT, SURF, EBBE, INMS and Cuckoo, respectively. However, in the 6th sequence, the values are 0.9, 0.87, 0.71, 0.79, 0.88, 0.87 and 0.83 for FB²S, MAA, SIFT, SURF, EBBE, INMS and Cuckoo, 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, 0.91, 0.78, 0.82, 0.92, 0.9 and 0.87 for FB²S, MAA, SIFT, SURF, EBBE, INMS and Cuckoo, 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 FB²S, MAA, SIFT, SURF, EBBE, INMS and Cuckoo, 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.

Table 6 the Results of our Proposed Algorithm

	A	B	C	D	P	R	Acc	E	F1
SeQ#1	46	4	44	6	0.92	0.89	0.9	0.1	0.91
SeQ#2	46	4	48	2	0.92	0.96	0.94	0.06	0.94
SeQ#3	48	2	46	4	0.96	0.92	0.94	0.06	0.94
AVG (3-SeQs)	46.7	3.3	46	4	0.93	0.92	0.93	0.07	0.93
SeQ#4	47	3	45	5	0.94	0.9	0.92	0.08	0.92
SeQ#5	49	1	47	3	0.98	0.94	0.96	0.04	0.96
AVG (5-SeQs)	47.2	2.8	46	4	0.94	0.92	0.93	0.07	0.93
SeQ#6	45	5	45	5	0.9	0.9	0.9	0.1	0.9
SeQ#7	48	2	47	3	0.96	0.94	0.95	0.05	0.95
SeQ#8	48	2	45	5	0.96	0.91	0.93	0.07	0.93
AVG (8-SeQs)	47.1	2.9	45.9	4.1	0.94	0.92	0.93	0.07	0.93
SeQ#9	47	3	44	6	0.94	0.89	0.91	0.09	0.93
SeQ#10	49	1	45	5	0.98	0.91	0.94	0.06	0.94
AVG (10-SeQs)	47.3	2.7	45.6	4.4	0.95	0.92	0.93	0.07	0.93

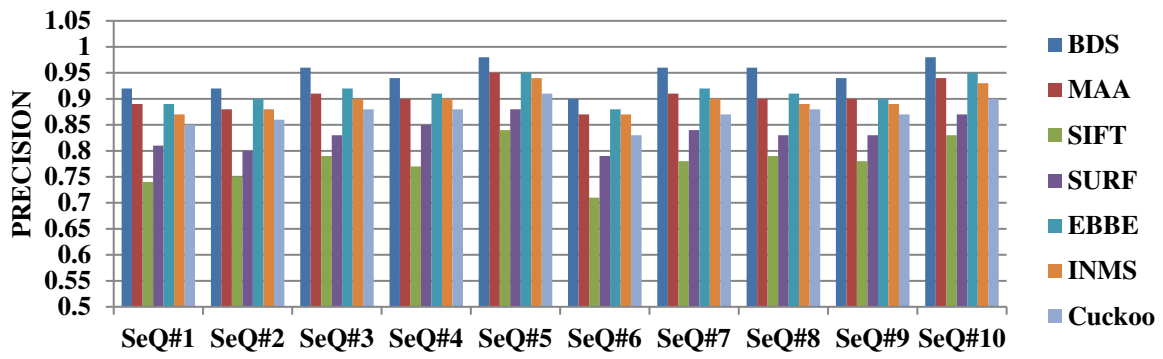


Figure 16 Precision of all Algorithms for the Ten Sequences

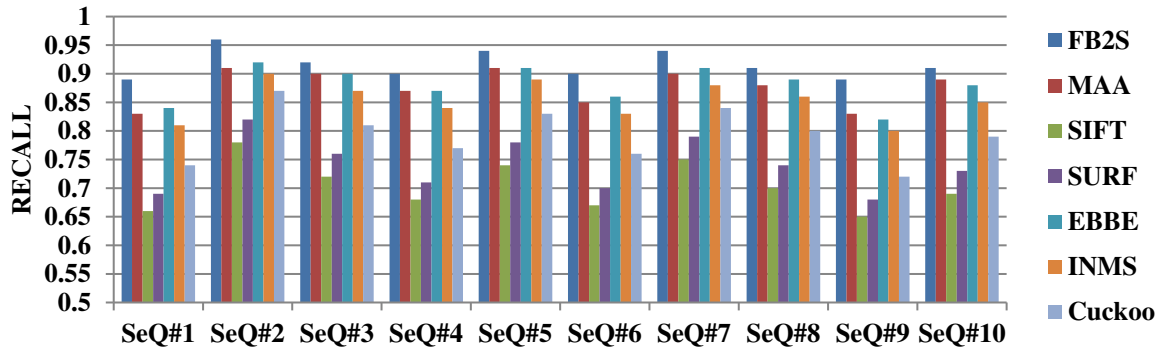


Figure 17 Recall of all Algorithms for the Ten Sequences

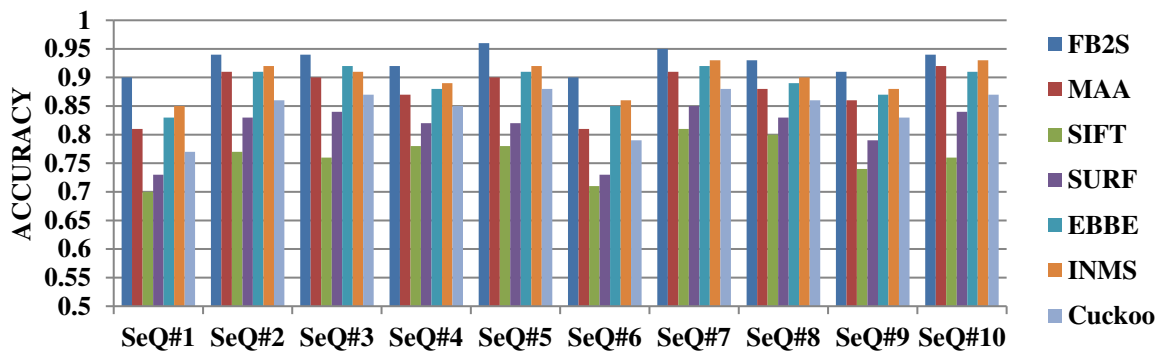


Figure 18 Accuracy of all Algorithms for the Ten Sequences

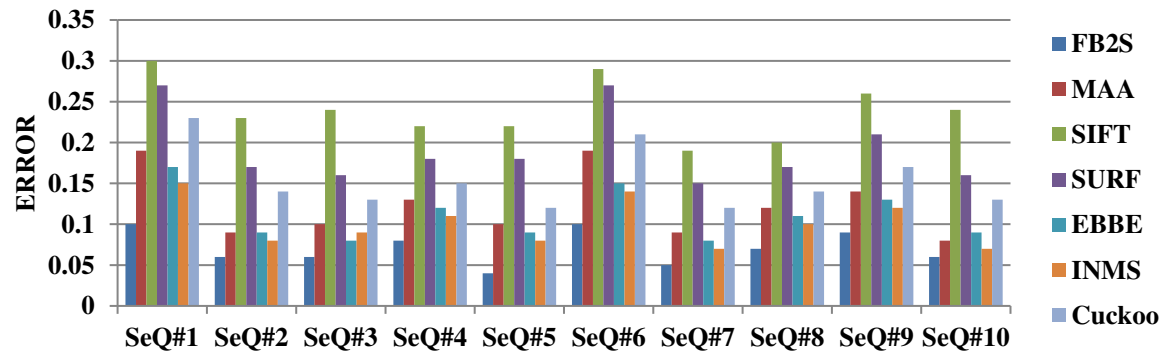


Figure 19 Error of all Algorithms for the Ten Sequences

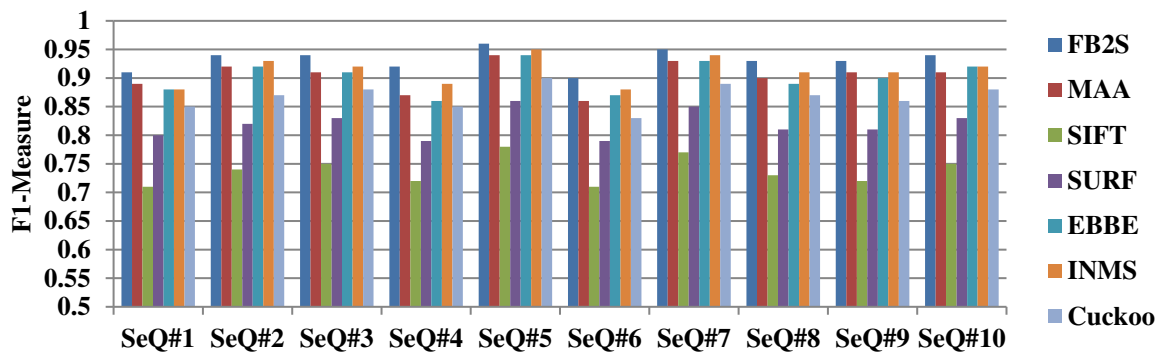


Figure 20 F1 Measure of all Algorithms for the Ten Sequences

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, 0.81, 0.7, 0.73, 0.83, 0.85 and 0.77, while the error values are 0.1, 0.19, 0.3, 0.27, 0.17, 0.15 and 0.23 for FB²S, MAA, SIFT, SURF, EBBE, INMS and Cuckoo, respectively. In sequence number 5, the accuracy values are 0.96, 0.9, 0.78, 0.82, 0.91, 0.92 and 0.88, but the error values are 0.04, 0.1, 0.22, 0.18, 0.09, 0.08 and 0.12 for FB²S, MAA, SIFT, SURF, EBBE, INMS and Cuckoo, respectively. Finally, the F1 measure values have been shown in Figure 20; in the 5th sequence, these values are 0.96, 0.94, 0.78, 0.86, 0.94, 0.95 and 0.9 for FB²S, MAA, SIFT, SURF, EBBE, INMS and Cuckoo, respectively. Furthermore, in the 6th sequence, the F1 values are 0.9, 0.86, 0.71, 0.79, 0.87, 0.88 and 0.83 for FB²D, MAA, SIFT, SURF, EBBE, INMS and Cuckoo, 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 its values in the first sequence.

In the current paper, we also find the average precision, average recall, average accuracy, average error and average F1-measure for 3, 5, 8, and 10 sequences. These results have been calculated using our proposed algorithm FB²S. These results approve that FB²S system is more reliable and perform perfectly. Figure 21 illustrated that average precision, average recall, average accuracy, average error and average F1-measure are 0.93, 0.92, 0.93, 0.07 and 0.93, respectively, when only three sequences are used, respectively. 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. 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, when the eight sequences are used. 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.

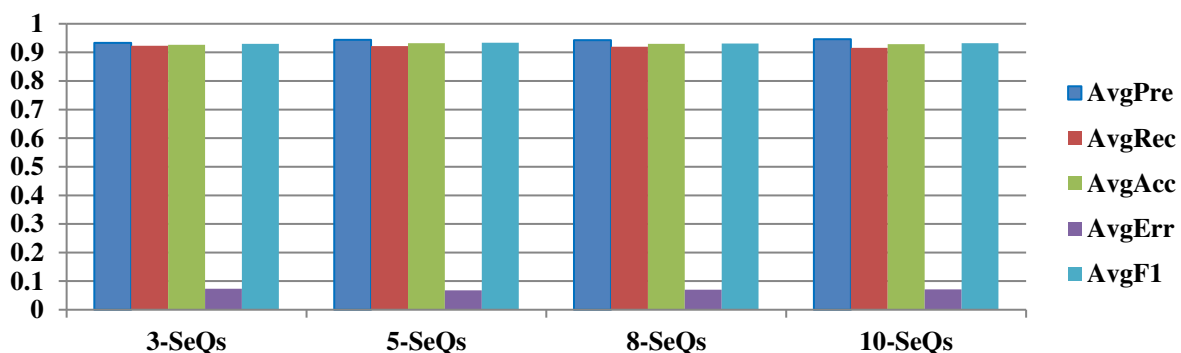


Figure 21 Average Precision, Recall, Accuracy, Error and F1-Measure for 3, 5, 8 and 10 Sequences

The experiments under study were conducted to various balls of different colors as shown in Figure 22. The proposed Fuzzy Based Ball Detection (FB²D) strategy can successfully detect balls with various colors.

Hence, the results of suggested algorithm, shown in Figures 16-20, prove that all balls are located in different positions on the field. The frame of the fast-moving ball is affected by motion blurring, as it is rendered larger in size and less in its white tone than the stationary or slower-moving ball. The proportion of white color within the object must never be less than 30% of the whole area [69]. All objects having color outside the previous percentage are excluded from further processing, while each remaining object is considered as a ball candidate.



Figure 22 Various Balls with different colors

Finally, as illustrated in the experimental results, it is shown that performance values of our proposed detection algorithm exceed the other competitors' detection algorithms in terms of precision, recall and accuracy. There are three reasons for this high results of our proposed detection algorithm. The first one is that the ball always has a circular shape because the camera is fixed orthogonally on the flying bee. However, the camera in the other competitors' algorithms is static and fixed at one side of the playground which makes the ball usually has distorted shapes; this reason makes it difficult to detect the ball professionally as it may have several shapes based on the camera projection angle. The second reason is that our flying bees effectively cover the whole area of the playground. On the other hand, most of the traditional detection algorithms are based on static cameras, which need complicated zooming and calibration techniques to efficiently detect the ball. Even techniques that rely on moving cameras still suffer from low detection efficiency due to the randomness movement of the ball. The third and final reason is that our proposed detection algorithm relies on simple computations in comparison to the traditional detection systems, and it guarantees a fast response of the bee. In addition, there is no delay for 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 Fuzzy Based Ball Detection (FB²D) strategy will be discussed. As depicted in Table 7, FB²S 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 via the H_Vector using the Scanner Bee. Thus, FB²S has the capability of on-line training and adaptive response. Besides, rules in the FLS 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 FB²S, fewer values, rules, and decisions are required, so the fuzzy logic system can be easily constructed and deployed. On the other hand, the proposed FB²S 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 FB²S are illustrated in table 7.

Table 7 Pros and Cons of our Proposed FB²S

FB ² S Pros		FB ² S Cons	
Property	Description	Property	Description
Efficient	<ul style="list-style-type: none"> ○ FB²S is efficient in detecting the playfield color as it constructs accurate range of green that represents the playing field. ○ FB²S employs an effective fuzzy inference engine (FIE) which pretends the way in which humans interpret linguistic values. ○ The employed FIE relates output to input, without having to understand all the variables, permitting the design of a system to be more accurate and stable. ○ FB²S 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]. 	Initialization time	Need some time before the match starts to detect the playfield color.
Adaptive	<ul style="list-style-type: none"> ○ 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. 	Expert guidance	<ul style="list-style-type: none"> ○ Fuzzy logic should be built with the complete guidance of experts. ○ Setting exact, fuzzy rules and, membership functions is a difficult task
Robust	<p>FB²S is a robust system as it employs fuzzy logic so that:</p> <ul style="list-style-type: none"> ○ 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. 	Occlusion	FB ² S has problems to deal with the ball when it is partially occluded.
Flexibility	<ul style="list-style-type: none"> ○ 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. 		
Fast	<ul style="list-style-type: none"> ○ 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 FB²S, 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	<ul style="list-style-type: none"> ○ The structure of Fuzzy Logic System, which is the core of the proposed FB²S, is easy and understandable. ○ In FB²S, fewer values, rules, and decisions are required. ○ FB²S can be easily constructed and deployed. 		
Reliable	<ul style="list-style-type: none"> ○ FB²S is reliable due to the engagement between the used ball detection technique and the hosting ball bee in a mutually beneficial relationship. 		

8. Conclusion

This paper introduced a method for the soccer ball detection using a quadcopter, which is called the Ball Bee. Our proposed Fuzzy Based Ball Detection (FB²D) Algorithm is divided into two main steps: (a) Background Identification, and (b) Object Detection. In the first step, the HSV color space had been used to find the hue values of the empty field before the game started. And then, these values in comparison to the corresponding values in the current frame of interest. On the other hand, the second step consists of four phases: (i) Background Subtraction, (ii) Edge detection, (iii) Ball Candidates Identification, and (iv) Classifier. In first phase, the matching hue values of the taken frame will be removed, and then the Canny edge detector has been applied in the second phase. In the third phase, the candidate ball from the objects founded in the frame has been identified using the Circle Hough Transform technique. Finally, in the fourth phase, a Fuzzy Inference System has been defined with three fuzzy sets and eight rules to decide which candidate is the actual ball according to specific membership functions. Experimental results showed that the detection of the soccer ball in real time is possible even if the ball is occluded by other objects or players.

The system is intended 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 FB²S consumes some time before the match starts, as the Scanner Bee takes a few frames of the empty playfield in order to determine its true color values. It also has problems to detect the ball when it's possessed or partially occluded by a player.

Conflicts of Interest Statement

The authors whose names are listed immediately below certify that they have NO affiliations with or involvement in any organization or entity with any financial interest (such as honoraria, educational grants, participation in speakers' bureaus, membership, employment, consultancies, stock ownership, or other equity interest, and expert testimony or patent-licensing arrangements), or non-financial interest (such as personal or professional relationships, affiliations, knowledge or beliefs) in the subject matter or materials discussed in this manuscript.

Authorship contributions

All authors contributed to the study conception and design. Material preparation, data collection and analysis were performed by all authors. The first draft of the manuscript was written by Arwa E. Abulwafa and all authors commented on previous versions of the manuscript. All authors read and approved the final manuscript. All authors contribute equally.

9. References

- [1] U. Rao and U. C. Pati, "A novel algorithm for detection of soccer ball and player", In International Conference on Communications and Signal Processing (ICCSP), IEEE, pp. 344-348, 2015.
- [2] G. Gomez, P. H. López, D. Link and B. Eskofier, "Tracking of ball and players in beach volleyball videos", PloS One, vol. 9, no. 11, 2014.
- [3] Y. Yoon, H. Hwang, Y. Choi, M. Joo, H. Oh, I. Park, K. Lee and J. Hwang, "Analyzing basketball movements and pass relationships using Realtime object tracking techniques based on deep learning.", IEEE Access, vol. 7, pp. 56564-56576, 2019.
- [4] V. Reno, N. Mosca, R. Marani, M. Nitti, T. D'Orazio and E. Stella, "Convolutional neural networks-based ball detection in tennis games", In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, pp. 1758-1764, 2018.
- [5] M. Pobar, and M. Ivasic-Kos, "Active Player Detection in Handball Scenes Based on Activity Measures", Sensors, vol. 20, no. 5, p. 1475, 2020

- [6] S. A. Velastin, R. Fernández, J. E. Espinosa, and A. Bay, “Detecting, tracking and counting people getting on/off a metropolitan train using a standard video camera”, *Sensors*, vol. 20, no. 21, pp. 6251, 2020.
- [7] C. Cuevas, D. Quilon and N. Garcia, “Techniques and applications for soccer video analysis: A survey”, *Multimedia Tools and Applications*, vol. 79, no. 39, pp. 29685-29721, 2020.
- [8] J. Yu, Y. Tang, Z. Wang, and L. Shi, “Playfield and ball detection in soccer video”, In *International Symposium on Visual Computing*, Springer, Berlin, Heidelberg, pp. 387-396, 2007.
- [9] P. Hansen, P. Franco and S. Y. Kim, “Soccer ball recognition and distance prediction using fuzzy Petri nets”, In *2018 IEEE International Conference on Information Reuse and Integration (IRI)*, pp. 315-322, 2018.
- [10] D. Meneghetti, T. Homem, J. de Oliveira, I. Da Silva, D. Perico and R. Bianchi, “Detecting soccer balls with reduced neural networks: a comparison of multiple architectures under constrained hardware scenarios”, *arXiv e-prints*, 2020.
- [11] B. Zhou, H. Koerger, M. Wirth, C. Zwick, C. Martindale, H. Cruz, B. Eskofier and P. Lukowicz, “Smart soccer shoe: monitoring foot-ball interaction with shoe integrated textile pressure sensor matrix”, In *Proceedings of the 2016 ACM International Symposium on Wearable Computers*, pp. 64-71. 2016.
- [12] M. Heryanto, H. Suprijono, B. Suprpto and B. Kusumoputro, “Attitude and altitude control of a quadcopter using neural network based direct inverse control scheme”, *Advanced Science Letters*, vol. 23, no. 5, pp. 4060-4064, 2017.
- [13] A. Abulwafa, A. Saleh, H. Ali and M. Saraya, “A fog-based ball tracking (FB2T) system using intelligent ball bees”, *Journal of Ambient Intelligence and Humanized Computing*, vol. 11, no. 11, pp. 5735-5754, 2020.
- [14] N. Joshi, “4 sensors that are being used in drones |IOT| Drones Technology”, 2016. (accessed February 1, 2020).
- [15] L. Brown, “What is a Drone and How Does it Work”, 2019. (accessed February 1, 2020).
- [16] N. Ishii, I. Kitahara, Y. Kameda and Y. Ohta, “3D tracking of a soccer ball using two synchronized cameras”, In *Pacific-Rim Conference on Multimedia*, Springer, Berlin, Heidelberg, pp. 196-205, 2007.
- [17] J. Ren, J. Orwell, G. A. Jones and M. Xu, “Tracking the soccer ball using multiple fixed cameras”, *Computer Vision and Image Understanding*, vol. 113, no. 5, pp. 633-642, 2009.
- [18] V. Pallavi, J. Mukherjee, A. K. Majumdar, S. Sural, “Ball detection from broadcast soccer videos using static and dynamic features”, *Journal of Visual Communication and Image Representation* vol. 19, no. 7, pp. 426-436, 2008.
- [19] X. Yu, H. Leong, C. Xu and Q. Tian, “Trajectory-based ball detection and tracking in broadcast soccer video”, *Multimedia, IEEE Transactions*, vol. 8, no. 6, pp. 1164-1178, 2006.
- [20] T. D’Orazio, C. Guaragnella, M. Leo and A. Distante, “A new algorithm for ball recognition using circle hough transform and neural classier”, *Pattern Recognition*, vol. 37, no. 3, pp. 393-408, 2004.
- [21] P. Mazzeo, M. Leo, P. Spagnolo and M. Nitti, “Soccer Ball Detection by Comparing Different Feature Extraction Methodologies”, *Advances in Artificial Intelligence*, 2012.
- [22] M. Polceanu, A.-O. Petac, H. Lebsir, B. Fiter, and C. Buche, ‘Real time tennis match tracking with low-cost equipment’, In *FLAIRS*, vol. 31, pp. 197–200, 2018.
- [23] F. Yan, W. Christmas and J. Kittler, “A tennis ball tracking algorithm for automatic annotation of tennis match”, In *British machine vision conference*, vol. 2, pp. 619–628, 2005.
- [24] N. Owens, C. Harris, and C. Stennett, “Hawk-eye tennis system”, In *Visual Information Engineering, International Conference*, pp. 182–185, 2003.

- [25] C. Conaire, P.Kelly, D. Connaghan and N. O'Connor, "Tennissense: A platform for extracting semantic information from multi-camera tennis data", In Digital Signal Processing, 16th International Conference, IEEE, pp. 1-6. 2009.
- [26] M. Fazio, K. Fisher and T. Fujinami, "Tennis ball tracking: 3d trajectory estimation using smartphone videos", Department of Electrical Engineering, Stanford University, 2018.
- [27] M. Archana and M. Geetha, "Object detection and tracking based on trajectory in broadcast tennis video", *Procedia Computer Science*, vol. 58, pp. 225-232, 2015
- [28] K. Teachabarikiti, T. Chalidabhongse and A. Thammano, "Players tracking and ball detection for an automatic tennis video annotation", In Control Automation Robotics & Vision (ICARCV), 11th International Conference, IEEE, pp. 2461-2494., 2010.
- [29] M. Huang and C.-p. Huang, "An optimal image segmentation clustering algorithm for badminton sport moving tracking applications from video sequences", *IJCSIS*, vol. 6, no. 9, 2017.
- [30] J. Wang and N. Parameswaran, "Survey of Sports Video Analysis: research issues and applications", In Proceedings of the Pan-Sydney area workshop on Visual information processing, pp. 87-90. 2004.
- [31] C. Needham and R. Boyle, "Tracking multiple sports players through occlusion, congestion and scale", In: *BMVC*, vol. 1, no. 1, pp. 93-102, 2001.
- [32] T. D'Orazio, N. Ancona, G. Cicirelli and M. Nitti, "A ball detection algorithm for broadcast soccer image sequences", *IAPR ICPR*, vol. 2, 2002.
- [33] O. Nemethova, M. Zahumensky and M. Rupp, "Preprocessing of ball game video sequences for robust transmission over mobile network", In *CDMA International Conference (CIC)*, Seoul, 2004.
- [34] L. Fernandes and M. Oliveira, "Real-time line detection through an improved Hough transform voting scheme", *Pattern Recognition*, vol. 41, no. 1, pp. 299-314, 2008.
- [35] X. Yu, C. Xu, Q. Tian, H. Leong, "A ball tracking framework for broadcast soccer video", In proceeding of International Conference on multimedia and expo. ICME'03. (Cat. No. 03TH8698), vol. 2, pp. II-273. IEEE, 2003.
- [36] D. Liang, Y. Liu, Q. Huang, and W. Gao, "A scheme for ball detection and tracking in broadcast soccer video", In *Pacific-Rim Conference on Multimedia*, pp. 864-875. Springer, Berlin, Heidelberg, 2005.
- [37] Y. Liu, S. Jiang, Q. Ye, W. Gao and Q. Huang, "Playfield Detection Using Adaptive GMM and Its Application", In *Proceeding of IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP'05*, vol. 2, pp. 421-424, 2005.
- [38] X. Tong, H.Q. Lu and Q.S. Liu, "An Effective and Fast Soccer Ball Detection and Tracking Method", In *Proceeding of the 17th International Conference on Pattern Recognition (ICPR)*, Cambridge, UK. pp. 1-4, 2004.
- [39] O. Utsumi, K. Miura, I. Ide, S. Sakai and H. Tanaka, "An object detection method for describing soccer games from video", In *Proceeding of IEEE International Conference on Multimedia and Expo*, vol. 1, pp. 45-48, 2002.
- [40] A. Manzanera and J. Richefeu, "A new motion detection algorithm based on R-D background estimation", *Pattern Recognition Letters*, vol. 28, no. 3, pp. 320-328, 2007.
- [41] M. Oral and U. Deniz, "Center of mass model: A novel approach to background modeling for segmentation of moving objects", *Image and Vision Computing*, vol. 25, pp. 1365-1376, 2007.
- [42] Z. Fu and Y. Han, "Centroid weighted Kalman filter for visual object tracking", *Journal of Measurement*, vol. 45, no. 4, pp. 650-655, 2012.
- [43] S. Weng, C. Kuo and S. Tu, "Video object tracking using adaptive Kalman filter. *Journal of Visual Communication and Image Representation*, vol. 17, no. 6, pp. 1190-1208, 2006.

- [44] Li. Yao, M. Ling, “An Improved Mixture-of-Gaussians Background Model with Frame Difference and Blob Tracking in Video Stream”, *The Scientific World Journal*, pp. 1-9, (2014).
- [45] S. Kumar and J. Yadav, “Background Subtraction Method for Object Detection and Tracking”, In *Proceeding of international conference on intelligent communication, control and devices*, Springer, Singapore, pp. 1057-1063, 2017.
- [46] Vo, G.D.; Park, C. “Robust regression for image binarization under heavy noise and nonuniform background”, *Pattern Recognition*, vol. 81, pp. 224-239, 2018.
- [47] F.-C. Cheng, S.-C. Huang and S.-J. Ruan, “Advanced motion detection for intelligent video surveillance systems”, In *Proceedings of the 2010 ACM Symposium on Applied Computing*, pp. 983-984. 2010.
- [48] Q.-Z. Li, D.-X. He, B. Wang, “Effective Moving Objects Detection Based on Clustering Background Model for Video Surveillance”. In *Proceedings of the Congress on Image and Signal Processing*, Sanya, Hainan, China, vol. 3, pp. 656-660. IEEE, 2008.
- [49] C. Chiu, M. Ku and L. Liang, “A robust object segmentation system using a probability-based background extraction algorithm”, *IEEE Transactions on circuits and systems for video technology*, vol. 20, no. 4, pp. 518-528, 2010.
- [50] T. D’Orazio, C. Guaragnella, M. Leo and A. Distanto, “A new algorithm for ball recognition using circle Hough transform and neural classifier”, *Pattern Recognition*, vol. 37, no. 3, pp. 393-408, 2004.
- [51] U. Moyyila, “Detection and Recognition of Soccer Ball and Players”, PhD dissertation, 2015.
- [52] S.-Y. Chiu, C.-C. Chiu, and S. Xu, “A Background Subtraction Algorithm in Complex Environments Based on Category Entropy Analysis”, *Applied sciences*, vol. 8, no. 6, pp. 885, 2018,
- [53] A. Bondalapati, S. Bhavanam and E. Reddy, “Moving object detection based on unified model”, *Journal of Ambient Intelligence and Humanized Computing*, pp. 1-16, 2020.
- [54] S. Balaji, S. Karthikeyan and R. Manikandan, “Object detection using Metaheuristic algorithm for volley ball sports application.” *Journal of Ambient Intelligence and Humanized Computing*, pp. 1-11, 2020.
- [55] Y. Huang, J. Llach and S. Bhagavathy, “Players and Ball Detection in Soccer Videos Based on Color Segmentation and Shape Analysis”, In *Proceedings of Multimedia Content Analysis and Mining (MCAM)*, Weihai, China. pp. 416-425, 2007.
- [56] M. Ali, M. Abdullah-Al-Wadud and S.-L. Lee, “An Efficient Algorithm for Detection of Soccer Ball and Players”, In *Proceeding of Conference on Signal and Image Processing (SIP)*, vol. 16, pp.1-8, 2012.
- [57] P. Mudjirahardjo, M. Purnomo, R. Hasanah and H. Suyono, “Multi Scale Performance of Feature Extraction for Human Head Recognition”, *International Journal of Applied Engineering Research*, vol. 10, no.17, pp. 37731-37736, 2015.
- [58] P. Mudjirahardjo, M. Purnomo, R. Hasanah and H. Suyono, “Histogram of Transition for Human Head Recognition”, *Jurnal Teknologi*, vol. 78, no. 5-9, pp. 53-58, 2016.
- [59] B. Chakraborty and S. Meher, “A Trajectory-Based Ball Detection and Tracking System with Applications to Shooting Angle and Velocity Estimation in Basketball Videos”, In *Proceeding of Annual IEEE India Conference (INDICON)*, Mumbai. pp. 1-6, 2013.
- [60] M. Durus, “Ball Tracking and Action Recognition of Soccer Player in TV Broadcast Videos”, PhD Dissertation, Technische Universitat Munchen, 2014.
- [61] P. Mudjirahardjo, J. Tan, H. Kim and S. Ishikawa, “Abnormal Motion Detection in an Occlusive Environment”, In *Proceeding of SICE Annual Conference*, Nagoya-Japan. pp. 1398-1402, 2013.
- [62] P. Mudjirahardjo, J. Tan, H. Kim and S. Ishikawa, “Temporal Analysis for Fast Motion Detection in a Crowd”, *Journal Artificial Life and Robotics*, vol. 20, no.1, pp. 56-63, 2015.

- [63] P. Mudjirahardjo and P. Siwindarto, "Soccer Field Detection based on Histogram of S-RGB", *ARNP Journal of Engineering and Applied Sciences*, vol. 11, no. 21, pp. 12405-12408, 2016.
- [64] W. Chen, Y. Shi, and G. Xuan. "Identifying computer graphics using HSV color model and statistical moments of characteristic functions", In *2007 IEEE International Conference on Multimedia and Expo*, pp. 1123-1126, 2007.
- [65] I. Pitas, "Digital image processing algorithms and applications", John Wiley & Sons, 2000.
- [66] D. Marr and E. Hildreth, "Theory of edge detection", In *Proceeding of the Royal Society of London. Series B. Biological Sciences* vol. 207, no. 1167, pp. 87-217, 1980.
- [67] R. Kirti and A. Bhatnagar, "Image Segmentation Using Canny Edge Detection Technique", In *Proceeding of international journal of techno-management research*, pp. 8-14, 2017.
- [58] V. Yadav, S. Batham, A. Acharya and R. Paul, "Approach to accurate circle detection: Circular Hough Transform and Local Maxima concept", In *2014 International Conference on Electronics and Communication Systems (ICECS)*, IEEE, pp. 1-5, 2014.
- [69] J. Ren, J. Orwell, G. A. Jones and M. Xu, "Real-Time Modeling of 3-D Soccer Ball Trajectories from Multiple Fixed Cameras", In *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 18, no. 3, pp. 350-362, 2008.