Ball Automatic Detection and Tracking in Long Shot Views

Mohaddeseh Kia
M.Sc. in Computer Engineering, Artificial Intelligence, Iranian Social Security Organization, Golestan Province Health Administration

Abstract
This study presented a new method to recognize and track soccer ball as well as long shot view recognition from other views. Soccer ball recognition in the proposed method composed of two parts. Soccer balls initial recognition is done through comparative thresholding on grass, bubbles extraction and analysis. Recognition algorithm or cascade classifier and pseudo Har properties are used in the case of ball overlapping or obstruction with other bubbles occurred. Optical flow property was applied to decrease the search space, the ball is tracked by Locus Kanade method in the pyramid mode which not only identifies the ball more precisely in the candidate bubbles but also reduces the range of applying pattern recognition algorithm in the overlapped situation requiring less computational cost. The results of two films show that in long shot recognition the percentage of accurate recognition is higher than 99% and error percentage is less than 0.8%. Moreover, comparing ball recognition with two competing methods showed that the proposed method had more recognition number percent and the precision has been improved 9.43 and 0.46 % as compared to the two competing methods.
Keywords
Tracking; Ball; Soccer; Locus Kanade; Cascade classifier; Pseudo Har properties

I. Introduction

In recent years, developing soccer and its popularity among different communities made the executives to prevent referee errors in soccer matches through applying various technologies. High percentage frequently human errors occur in soccer matches may unfairly cause winning or losing of a team. In the case of applying technologies including soccer ball tracking or technology of recording players activity in the field, soccer would become a better and more fairly game and sport by preventing human errors. So, this study presented a new method to track soccer ball in the video.
Following soccer ball usually includes two ball identification and tracking stages. Some provided methods identify soccer ball by properties such as shape and color [1-4]. Others used colored histogram and morphology. They identified soccer ball in the set of videos in term of white color property and circular shape. Hough transform method with its modifications is used as robust methods to identify circle then the ball.
[5] used Hough transform for ball recognition. However, it merely discovered the ball with no tracking. A revised version of circular Hough transform to identify soccer ball was applied in [6]. Other studies also employed pattern adaptation methods in ball recognition. [7] used pattern adaptation (matching) in order to identify the ball in different intervals and images; then, the ball is tracked within these intervals.
The method of Cascade classifier and Pseudo- Har properties is regarded as one of the most successful algorithms in object detection particularly face, face components and objects like logos [8-9]. Viola and Jones initially applied the method in object detection by using two important adjustments: 1. a cascade classifier, 2. Self-running. Standard AdaBoost classifier measures all weak classifiers in its total set to train and select property; then, produces a decision as outcome. This strategy of training is extremely slow; however, allocation process will quickly be done following training.
There have been provided various methods for soccer ball tracking including Calman filter method. In [2] the ball is first identified using size, color and shape; then, tracked through the information obtained from Calman filter. In another study [10] the ball is tracked using pattern match and Calman filter when no player is near the ball. When the ball is lost near the players, the search is initiated in the area around players till the ball is found. The drawback of this method is that when the ball is disappeared as the obstruction with players, Calman filter is unable to perfectly predict ball location, especially if the player holds the ball for a long time; thus, the error may increase. Some methods, also, applied neural networks and Bayesian methods in soccer ball tracking. [11] used Bayesian estimation framework requiring some cameras. In [12] the detected ball was modified by Hough transform and tracked through neural network.
The other concerns in automatic ball tracking are suddenly variation of video scenes. In above methods usually long shot images are manually segregated, then sent to algorithm. But, the proposed method in this paper allow the possibility of detecting long shot images from other images like close up or audience images. Previous view detection methods [13] require some other process in addition to ball detection and tracking which lead to total speed reduction while the necessary information are used, in the suggested method, to detect ball as well as view.

Manuscript received June 5, 2016
Manuscript revised June 20, 2016

There are seen challenges and drawbacks in most mentioned papers some of which are occlusion or overlapping and high computational complexity problems. The method stated in [1] is sensitive to overlapping, the problem is also seen in [5] and [6] with no resistance to occlusion. Moreover, in addition to occlusion in [2] it is inefficient in ball sudden movements. Occlusion makes problem in method of [10] particularly when the player holds the ball for a long time the error may more increase.
Ball recognition process, in the proposed method, is composed of initial detection and detection within overlapping. In this study adaptive threshholding was used in initial detection to extract bubbles (including ball, player, barriers or grass lines) and the ball was detected through analyzing these bubbles. In the case of ball overlapping or occlusion with other bubbles recognition algorithm or cascade classifier and pseudo- Har properties are used. But it must be noticed that applying this transform in the whole image is a long, time consuming process; therefore, to reduce search space the optical flow property with Lucas Kanade pyramid method were utilized in ball tracking. In this method the ball is detected more precisely among candidate bubbles; the range of applying pattern recognition algorithm may be reduced and it requires less computational costs. In the following, the proposed method and how to implement each part are presented in section 2. Section 3 showed adjusting the proposed model as well as results.

II. Proposed Algorithm

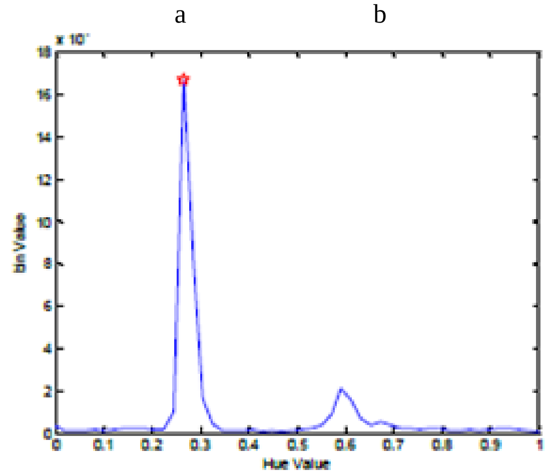
This thesis used 6 total parts to reveal and track soccer ball in video sequences.

1. Detecting grass range and view type (including long shot, close up, audience, and other various views). However, the goal of this part is just detecting long shot views.
2. Applying color filter for extracting bubbles on the grass field (including players, ball, barriers, lines as well as any objects with different color to grass but is in the grass range.
3. Classifying bubbles in to bubbles look like ball and bubbles with no ball like characteristics (players and lines). This part is useful to increase computations speed or in ball overlapping with other bubbles.
4. Implementing pyramid Lucas Kanade
5. Detecting overlap.
6. Implementing ball detection algorithm based on adaboost (this part consisted of training data, adaboost classifier training, ball diagram block.

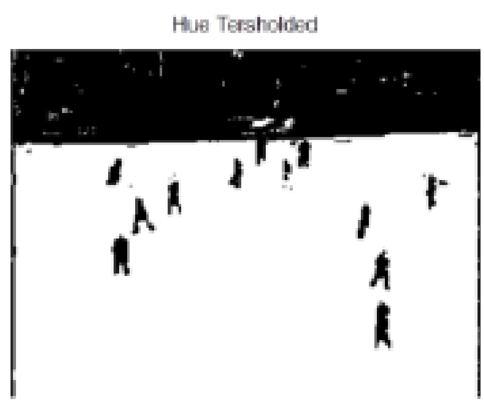
A. Detecting grass range and view type

Grass is easily detected through a simple thresholding but the problem is the variation of grass color in different matches or even the grass green color is extremely different from other parts of the field due to shadow or intense sunlight; applying a homogenous global threshold would disorganize extracting objects in the field. Two approaches are introduced to overcome this problem. First, transforming color space to HSV which H part contains color and is strongly resistant to lightening intensity. Second green color range is determined at any moment by aid of H image histogram; then, grass border is obtained through thresholding and morphology operation. How to detect grass border is presented in Fig 1.

a b



c



d

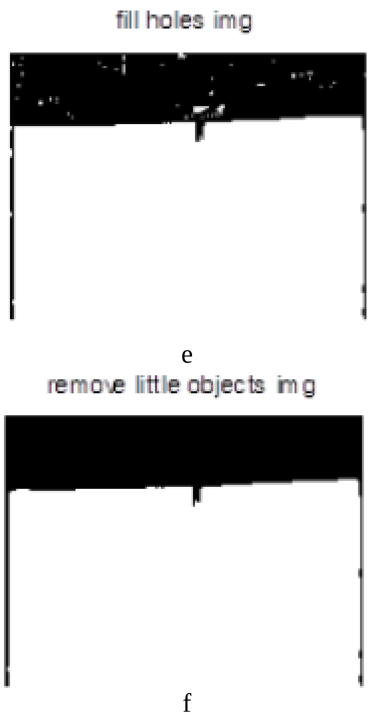


Figure 1. How to estimate grass border, a: input image, b: hue transform, c: hue histogram and the amount of grass hue, d: thresholding on hue page, e: filling the bubbles in the grass, f: selecting the largest area and removing unwanted margins.

B. View type recognition

As our purpose is just recognizing long shot; and on the other, the field borders have been determined to recognize bubbles, it is only required to frequently monitor field shape till the ratio of this mass to the whole image is more than 0.5, the view will be long shot provided that field shape is permanently square or rectangular. Field shape integration was analyzed by using compression ratio. This ratio equals to bubble area divided by polygon area embedded on bubble. If the coefficient is less than 0.88, it means the grass is not integrated and probably the view will be regarded as close up. Figure 2 shows how the view type is identified.

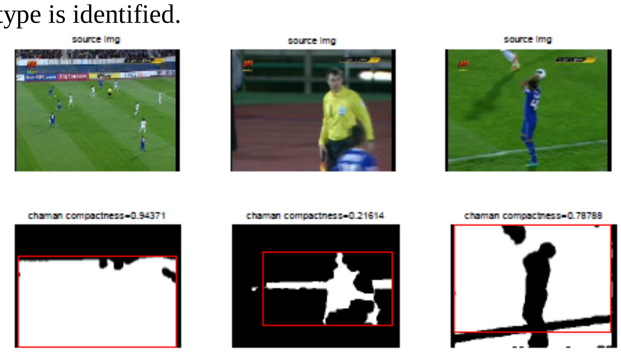


Figure 2. An example of detecting all various views using the proposed method

C. Applying shape and color filter for extracting bubbles on the grass field

As it is seen in Fig 2.a, there have been many bubbles detected in the grass. In this step, bubbles less looking like the ball must be removed; so it needs establishing some general rules to limit the candidate bubbles. In the case of overlapping, all bubbles must be checked through more precisely adaboost matching pattern to find the ball.
Some rules to reduce the number of candidates in term of bubble shape

1. Ball bubble has almost equal length and width (circularity property)
2. The bubble is the white colored- ball.
3. The ball size is in a certain range.

Whereas, the problem is that the ball is not always circular and may transform in to cylinder or oval shape.
[4] limited the ratio of length to width of the rectangular tangent on each bubble within 0.5 to 2 which was also used in this paper. The ball radius was measured 5 to 20 after analyzing ball radius in several videos.
In [3] ball color was recognized by the following formula.

$$B(x,y)=\begin{cases}1, & (r(x,y)-1/3)^2+(b(x,y)-1/3)^2\le a^2 \wedge I(x,y)\ge b\\ 0, & \text{otherwise}\end{cases} \qquad (1)$$

b(x,y), r(x,y) are blue and red components; I is the brightness; and a,b are two fixed thresholds. a=0.05 and b=160. Figure 3 shows filtering of ball candidates by help of size, colored filter as well as shape filter.

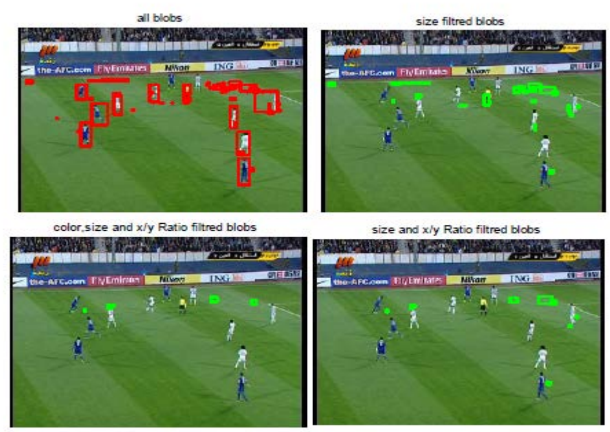


Figure 3. An instance of ball candidate bubbles by using size colored and shape filters.

D. Detecting overlaps

Occurring overlaps make detecting bubbles with features like ball impossible in the estimated location; in this way, other bubbles around the estimated point must be analyzed independent of bubble shape. The goal of ball tracking is to reduce search space so that decreasing computational cost since adaboost is considered as a time consuming method with high computational cost; hence reducing the size of the image this transform is applied may be very useful.

E. Har properties and cascade classifier in ball detection during overlap

There have been offered various methods for pattern matching. The goal of our application is ball detection, but algorithm high speed is considered important as the final algorithm must own high processing ability; thus, selecting most high computational methods may be impossible. One of the fast and successful methods in face, hand as well as human body recognition is applying Har like properties and combining these features to adaboost classifier [9-10]. These references used Fig 4 operators in shape properties extraction. As it can be seen, these operators composed of three general classes.

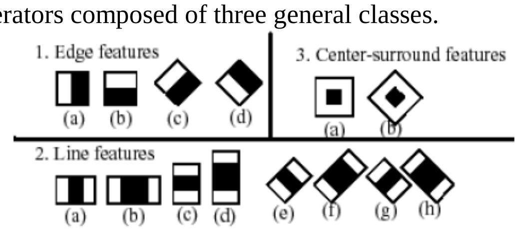


Figure 4. Structures of properties extraction

To extract the properties corresponded to each structure in Fig 4, image total pixels in the dark area were subtracted the total pixel of bright area, and the result will be used as the property. Clearly, these properties are too weak unable to highly distinguish. The purpose of creating these properties is computing a scalable and fast property.

1) Adaboost classifier

In the following, the properties were classified by adaboost classifier. This classifier, at the first step, selects the appropriate property for a pattern; and at second step, trains these properties. Generally, the goal of adaboost algorithm is to boost and increase the efficiency of simple classifying algorithms. These simple algorithms are so called “weak classifiers”. In first step, consider pattern matching in terms of a 24×24 pattern properties. According to the above structures, 45369 values can be extracted as properties [9]. Then, per each image sub-window being checked whether a pattern is or not, as much properties as this value may be extracted. Certainly, this number may never be implemented in a fast, high speedy application. The significance of adaboost algorithm relies on choosing a limited number of these properties according to pattern structure. The properties are extracted during training process and those with less change for al training data will be selected. Of the chosen properties, those with the highest distinction between two classes would be extracted as the final properties.

Assume that a weak classifier is presented by ($h_j(x)$) consisting of (f_j) property, (θ_j) threshold and (p_j) coefficient. Balance factor of setting direction has an unequally impact.

$$h_j(x) = \begin{cases} 1 & \text{if } p_j f_j(x) < p_j \theta_j \\ \text{otherwise} & \end{cases} \qquad (2)$$

Where, x is a pixel 24×24 sub-window of image.

Adaboost algorithm is explained as follows.

Assume that training data are presented by $(x_1, y_1), \ldots, (x_n, y_n)$ pairs. Y is the data label. If the window contains a pattern, it equals 1; otherwise 0.

The initial weight for y_i=0.1can be considered as $w_{1,i} = \frac{1}{2m}, \frac{1}{2l}$. M is the negative instances and 1 shows the positives. The weights are scaled for t=1,…,T.

$$w_{t,i} \leftarrow \frac{w_{t,i}}{\sum_{j=1}^{n} w_{t,i}}$$

The classifier (h_j) for each property (j) will be trained just with one property. The correspondent error to w_t is obtained by the following.

$$\varepsilon_{1,i} = \sum w_i \left| h_j(x_i) - y_i \right|. \qquad (3)$$

The least error classifier will be selected.

The weights are updated. $w_{t+1,i} = w_{t,i} \beta_t^{1-e^i}$, where e^i will be zero here if the sample is correctly classified, otherwise it is 1 and $\beta_t = \frac{\varepsilon_i}{1-\varepsilon_i}$

Finally, classifier output:

$$h(x) = \begin{cases} \sum_{t=1}^{T} \alpha_t h_t(x) \geq \frac{1}{2} \sum_{t=1}^{T} \alpha_t \\ \text{otherwise} \end{cases} \qquad (4)$$

And

$$\alpha_t = \log \frac{1}{\beta_t}$$

In some cases to increase detection speed, a series connection (cascade) of classifiers can be regarded [10]. These cascade classifiers accurately recognize pattern whereas there is still wrong detections (about 40%). Figure 4 illustrates the structure of this new classifier. With this mechanism, if sub-window contains no pattern, classifier quickly recognizes lack of pattern.

2) Ball detection by Adaboost

Pattern match, previously mentioned, is extremely dependent on training and the variety of training data. The goal, here, is ball detection. Database must be provided, before training, in advance. Regarding current imitations, 100 windows containing ball were manually selected of 100 initial image frames. All these areas were written in

16×16 pixels size. Then, to increase training data, 3 software images were produced based on each sub-image. For this purpose, brightness level is randomly changed, Gaussian noise will be applied; then, randomly revolved. Ultimately, 300 training images will be provided all of which are positive. Negative images are produced by 50 frames including ground, audiences as well as players with no ball. The mentioned images will be divided in to 16×16 pixels windows in order to produce negative database, each window is regarded as a negative training image. To increase training resistance, these images are noised like previous images. Finally, adequate training images will be obtained. Some training data bases are shown in Fig 5.

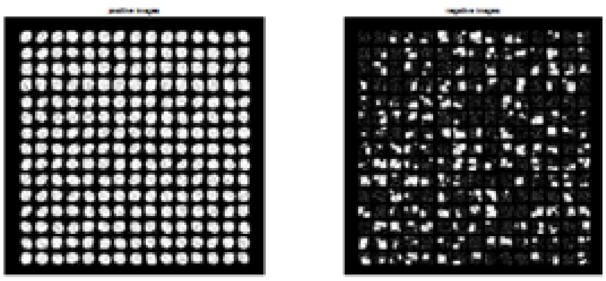


Figure 5. Data bases

Matlab functions, version 2013, were employed for training. The cascade classes were 10 and training windows equaled 14×14 pixels (must be smaller than training images); and maximum wrong detection was at least 50%. It is interesting that classifier may get to the required precision just in 5 classes and training will terminate. Since algorithm operates on gray scale image, if there is no color filter or field boundary filter being applied, it can detect all circular shapes. But by applying color filter and limiting replies to the inside grass boundaries (2-3 section) the following result is attained.

a

b

Figure 6. Ball detection through adaboost. a: without color filter, b: with color filter

The only problem is the algorithm high computation which must be limited by reducing search space in to possible locations determined by pyramid Lucas Kanade method.

F. Applying pyramid Locuas Kanade in estimating ball location in the following frames

Tracking is referred to estimation of the subject movement. There are various definitions in computer literature. In this paper, tracking is considered as estimating subject movement within consecutive frames. Due to ball fast movement it may happen during the game that ball travel a broad range in the distance between two consecutive frames; in other words, following ball movement path in consecutive frames it may be possible to estimate the ball possible location in the next frame. Optical flow and Lucas Kanade method are regarded as methods of tracking and tracing subjects. Suppose that two consecutive frames of A and B with δt interval are assumed in one scene. Movement of some parts of image pixels which is the considered object can be determined by using Lucas Kanade method.

- **Lucas Kanade**

To compute Lucas Kanade in a series images, the computations are based on two basic assumptions: first, bright intensity of the moving subject may not change in two consecutive images; secondly, movement of neighboring points of a moving object is like the movement of that point. In other word, speed amount would not suddenly change. Basic equations and Lucas Kanade computations in movement estimation, referring in multiple references, are summarily mentioned here.

Generally, $I_x u + I_y v + I_t = 0$ indicates the main equation of Lucas Kanade; where, I_v, I_x are horizontal and vertical gradients, respectively; and I_t time gradient of brightness level; u,v are horizontal and vertical optical flows, respectively [14]. To solve the above equation regarding that current papers used Lucas Kanade method in ball tracking [3], we also used this method. This method mainly focuses on finding movement speed in a way to minimize the following relation.

$$\epsilon(\nu) = \sum_{x=p_x-\omega_x}^{p_x+\omega_x} \sum_{y=p_y-\omega_y}^{p_y+\omega_y} \left(A(x,y) - B(x+\nu_x, y+\nu_y)\right)^2 \quad (5)$$

Given the proximity of consecutive images, the horizontal and vertical gradients will be as follows:

$$I_x(x,y) = \frac{\partial A(x,y)}{\partial x} = \frac{A(x+1,y) - A(x-1,y)}{2} \quad (6)$$

$$I_y(x,y) = \frac{\partial A(x,y)}{\partial y} = \frac{A(x,y+1) - A(x,y-1)}{2} \quad (7)$$

By deriving above equation for minimizing, the following relation is obtained. In other words, (3-4) relation is a value function based speed which equals zero if the speed is correctly estimated.

$$\begin{aligned} \nu_{optimal} &= G^{-1} b \\ G &= \sum_{x=p_x-\omega_x}^{p_x+\omega_x} \sum_{y=p_y-\omega_y}^{p_y+\omega_y} \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix} \\ b &= \sum_{x=p_x-\omega_x}^{p_x+\omega_x} \sum_{y=p_y-\omega_y}^{p_y+\omega_y} \begin{bmatrix} \delta I I_x \\ \delta I I_y \end{bmatrix} \end{aligned} \quad (8)$$

So, measuring object movement speed it is just required to implement equation (8). It must be noticed that the obtained speed vector contains horizontal and vertical components i.e. computing gradient relative to x and y is the requirement of implementing (8). To do this, the gradient must be estimated on a window (ω relative to G,b) around the considered point. G and b values will be computed through measuring mentioned values, placed in (8); then, speed will be computed.

Figure 7. Estimating ball movement direction in seven consecutive frames by using Lucas Kanade and crossing overlap with the ground line.

TABLE I. Results of view detection in two different videos.

The frames wrongly detected as long shot	Frames accurately detected as long shot	Number of long shot frames	Total frame numbers	Match number
8	1042	1050	2000	1
6	1311	1324	2000	1

TABLE II. Comparing results of ball detection in two different videos

Method	Competition Num	Frames total numbers	The frames with visible ball	Number of frames with accurate ball detection	Precision (%)
Recommended	1	650	609	601	92.46
[3]	1	650	609	513	78.9
[13]	1	650	609	597	91.85
Recommended	2	719	689	682	94.46
[3]	2	719	689	598	89.1
[13]	2	719	689	677	94.16

III. Evaluating the Proposed Algorithm

The proposed algorithm ability was studied and compared by using the data of [13]. This study estimated the results of two matches. The first was UEFA final match in 2003, Manchester United vs. Real Madrid; and the next was semi final UEFA between Chelsea and Arsenal, in 2006.

A. Assessing view type recognition

In both games, a sequence of 2000 frames was arbitrary chosen (frame 18000 to 20000), and the number of long shots were manually determined. As Table I shows, accurate recognition percentage in both games was higher than 99% and wrong detection was less than 0.8%.

B. Evaluating ball detection in long shot views

To exactly compare algorithms abilities, only the frames used in [13] were experimented. The results of two different matches are summarized in Table II and Fig 8.

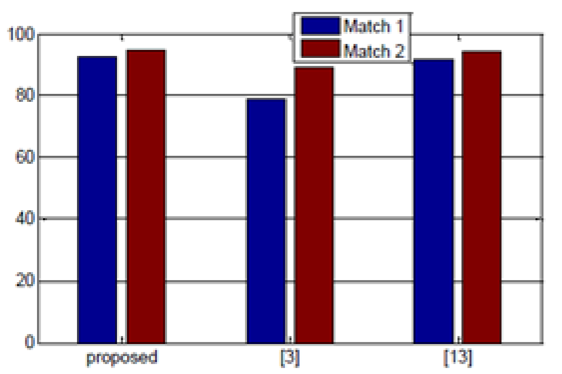


Fig 8. Three different methods results

Table II showed that the proposed method is much accurate than others, in which it has much higher detection number percentage. It has improved in the whole image 9.43% to [3] and 0.46% to [13], respectively. The algorithm average speed in a 6 GB RAM ad i5 CPU computer equals 10 f/s; but as speed is not specified in competing methods, this is not considered as a comparable parameter.

IV. Conclusion

This study presented a new method for soccer ball detection and tracking. Ball detection process, in the proposed method, consisted of two parts. Initial ball detection was done using adaptive thresholding and the bubbles in the grass were extracted. The ball will be detected by bubbles analysis. Recognition algorithm with cascade classifier or Har like properties is used in the case of overlapping or occlusion. The search space is reduced through optical flow feature with Locuas Kanade pyramid model in ball tracking. This method will detect the ball more precisely among bubbles; and also reduces the activity range of pattern recognition algorithm, if overlaps, requiring fewer computational costs. The results obtained from two different videos demonstrated that accurate detection in long shot is higher than 99% and wrong detection is less than 0.8%. Furthermore, comparing ball detection method to two competing methods showed that the proposed method has more percentage of detection number as compared to other methods, which relative to [3] to total image 9.43 % and to [13] 0.46% , has improved, respectively.

References

[1] Gong Y., Sin L. T., Chuan C. H., Zhang H. J., Sakauchi M." Automatic Parsing of TV Soccer Programs". In: Proceedings of Second International Conference on Multimedia Computing and Systems (1995),pp: 167-174.

[2] Yu X., Xu. C., Leong. H.W., Tian Q., Tang Q., Wan K.W., "Trajectory-based ball detection and tracking with applications to semantic analysis of roadcast soccer video", in: Proc. ACM Multimedia, 2013, pp: 11–20.

[3] Dawei L., Yang L., Qingming H., Wen G., "A Scheme for Ball Detection and Tracking in Broadcast Soccer Video" , Advances in Mulitmedia Information Processing PCM ,Vol: 3767,2005, pp: 864-875.

[4] Y. Huang, J. Llach, S. Bhagavathy, "Players and Ball Detection in Soccer Videos Based on Color Segmentation and Shape Analysis", MCAM 2011, LNCS 4577, pp. 416–425.

[5] T. D’Orazio, N. Ancona, G. Cicirelli, M. Nitti, “ A Ball Detection Algorithm for Real Soccer Image Sequences”, in ICPR, 2002, pp: 201-213.
[6] J. Hossein-Khani, H. Soltanian-Zadeh, M. Kamarei and O. Staadt “Ball Detection With the aim of Corner Event Detection in Soccer Video”, Ninth IEEE International Symposium on Parallel and Distributed Processing (2011)
[7] .S Hinterstoisser, V Lepetit, S Ilic, P Fua & N Navab. Dominant Orientation Templates for Real-Time Detection of Texture-LessObjects. Conference on Computer Vision and Pattern Recognition, 2010.
[8] Paul Viola and Michael J. Jones, “Rapid Object Detection using a Boosted Cascade of Simple Features.”, IEEE CVPR, 2001.
[9] Rainer Lienhart and Jochen Maydt. An Extended Set of Haar-like Features for Rapid Object Detection. IEEE ICIP 2002, Vol. 1, pp. 900-903, Sep. 2002
[10] Seo Y., Choi S., Kim H., Hong K.S., “Where are the ball and players?: soccer game analysis with color based tracking and image mosaic” in: Proc. ICIAP, 1997, pp: 196–203.
[11] Toshihiko M., Atsushi M., Masahide N., Mahito F., Nobuyuki Y., “DISTRIBUTED PARTICLE FILTERING FOR MULTIOCULAR SOCCER-BALL TRACKINg”, In: Proc. ICASSP 2007, vol. 3, pp. 937–940.
[12] T. D’Orazio, C. Guaragnella, M. Leo, and A. Distante, “A new algorithm for ball recognition using circle Hough transform and neural classifier,” , Pattern Recognit., vol. 37, no. 3, pp. 393–408, 2004.
[13] J6. Pallavi, V., Mukherjee, J. Majumdar, A. K., and Sural, S: "Ball Detection from Broadcast Soccer Videos using Static and Dynamic Features", Journal of Visual Communication and Image Representation, Elsevier Pub, 2008.
[14] Barron, J.L., D.J. Fleet, S.S. Beauchemin, and T.A. Burkitt. Performance of optical flow techniques. CVPR, 1992.