J Opt (April–June 2013) 42(2):156–170
DOI 10.1007/s12596-012-0108-7

RESEARCH ARTICLE

A real-time trajectory-based ball detection-and-tracking framework for basketball video

Bodhisattwa Chakraborty · Sukadev Meher

Received: 4 January 2012 / Accepted: 12 December 2012 / Published online: 10 January 2013

Abstract In a sports video, the significant events are caused mostly because of ball-player and player-player interaction. To detect and track a ball or a player in a sports video becomes more challenging in presence of many moving objects in the background. The small size of the ball in relation to the frame size makes ball detection much more difficult. In addition, the ball-images are also getting deformed due to the high speed movement of the ball. Often the ball gets occluded by players and the ball image gets merged with lines and boarders in the field. In this paper, the problem of ball detection-and-tracking in a real time basketball video is addressed. Here a trajectory based ball-detection-and-tracking method is proposed to detect and track the ball in a set of videos of basketball long shot sequences. A two-fold detection-and-tracking framework is used where the ball is detected using a feature-based method in the first step. The second step verifies the detection result of the ball detection system using the 2D-trajectory information of the possible ball candidates in the frame. The true ball trajectory is extracted from a set of candidate trajectories and the ball locations are estimated along the trajectory. The missing ball positions due to occlusion of the ball and merging of the ball image with other objects in the background are estimated using a trajectory interpolation technique.

Keywords Ball detection-and-tracking · Sports video · Trajectory-based algorithm · Feature based · 2D trajectory · Trajectory growing · Trajectory interpolation

Introduction

The evolution in the digital equipments for recording and storing multimedia contents and the ever-increasing computing power of the digital computers opens up many research areas in the field of video processing and analysis. The analyses of sports video have created a lot of enthusiasm among researchers because it have rich multimedia contents and can be commercially harnessed. As the consumers are interested more in a set of specific events in a sport video, it has becomes a challenging research area to find out the interesting events in a sports video from a huge volume of available data. The research areas also include enhancing the presentation techniques for better viewing experience of the consumers. Now a day, sports video processing is extensively used for almost all types of sports. The key areas of sports video analysis revolves around video indexing and summarization, tactics inference, object tracking, computer

B. Chakraborty (✉) · S. Meher
Department of Electronics and Communication Engineering, National Institute of Technology Rourkela, Odisha 769008, India
e-mail: bodhisattwa.chakraborty@gmail.com

S. Meher
e-mail: sukadevmeher@gmail.com

assisted referring, shot classification, content insertion etc.

Video indexing and summarization is used in sports video to mark and retrieve a set of interesting events for the viewers based on high level semantic representations. These special events are usually known as *highlights*. The *highlights extraction* not only gives a shorter representation of the game to the viewers, it also helps coaches and players to analyse the game pattern for better preparation. Rui et al. [1] proposed an automatic highlight extraction system for broadcast baseball videos using the audio features. Assfalg et al. [2] presented a system for automatic annotations of highlights in soccer videos based on various visual cues such as ball motion, colour of the player's uniforms etc. Yow et al. [3] proposed an idea to analyse and present highlights from a soccer video by content analysis and panoramic reconstruction. Nepal et al. [4] used temporal models based on the pattern of occurrence of some key events during the game. The feature extracted based on these key events are used to identify the goal events in a basketball video. Li and Sezan [5] used a unified framework for event detection and summarization in sports video using low level visual features and rule based inference method.

The *tactics analysis* [6, 7] is hugely used by the players and the coaches for performance evaluation and enhancement in a sport. Using the image model of the tennis court line and the knowledge of the camera geometry used to capture the tennis video, Sudhir et al. [8] presented a system for automatic analysis of tennis video. Wang et al. [9] used trajectory information and landing position of a tennis ball to analyse the 58 winning patterns from broadcast tennis video clips. Taki et al. [10] proposed a motion analysis system to evaluate the teamwork in a soccer game based on movement of all the players in the game.

Object tracking is a useful aspect of sports video analysis. It is vastly used to detect and track players, balls or both ball and players in a sports video for various application and analysis. Liu et al. [11] presented a method of automatic detection, labelling and tracking of multiple players in a broadcast soccer video. Xing et al. [12] proposed a dual-mode two-way Bayesian inference approach with progressive modelling to track multiple highly dynamic and highly interactive players in various sports videos. It has been observed that to locate and track the ball in a sport video is of great help while the sports video is being analysed for further understanding. Thus many research works has been done to detect and track the ball in sports video in recent years [13–19]. Sometimes it is necessary to track both the ball and players in a sports video for further game specific analysis. Chen and Zhang [20] presented an algorithm to automatically rank the highlights for broadcast table tennis games. The highlights are ranked according to some extracted high level semantic features which includes the position of the table, the player action and the ball trajectory. Choi et al. [21] proposed a method of tracking the ball and the players in multiple soccer video sequences taken from fixed cameras located around the stadium. They used images from multiple cameras to eliminate the effect of occlusion between players in the same team. The ball tracking is achieved by using the results of player tracking.

Referee assistance system helps the referee to make crucial decisions based on real-time video analysis system. The Hawk-eye [22] system is being extensively used in cricket and tennis games. Hashimoto and Ozawa [23] proposed a system for automatic judgement of the offside in soccer games. The players are detected and tracked using multiple cameras and a formation analysis by classifying the uniforms and calculate the position of an offside line. The moments of passes are recognized using the 3D trajectory of the ball.

Shot classification is one of the fundamental works of sports video analysis. Duan et al. [24] proposed a unified framework for semantic shot classification in sports video employing supervised learning to perform a top-down video shot classification using mid-level representations. Hua et al. [25] employ scene classification technique for baseball TV broadcast videos using maximum entropy scheme along with multimedia features such as image, audio and speech. The image features include color distribution, edge distribution and camera motion. The players are searched using color, edge and texture information in the images. The closed caption feature is also integrated to classify the different scenes in the baseball video.

Content insertion in sports video serves a great deal of commercial benefits. The sports video is an excellent source of advertisement as it is being viewed by a large number of audiences. Thus it is necessary to customise the sports video by inserting and changing some captions and banners according to the need of the local audience. The challenging part of the content insertion lies in inserting the additional content at correct place and correct time without disturbing the actual content and ascent of the sports video. Wan et al. [26] proposed a virtual content insertion system in a soccer video by detecting the goalmouth. A Hough-transform based line-mark detection technique is used to detect the dominant green regions. The vertical and horizontal goal-bars are isolated using a colour-based region growing technique.

In this paper, a trajectory-based ball detection-and-tracking algorithm with trajectory interpolation to find out the missing ball location along the ball flight path in a basketball game video is proposed. To encounter with the problem of ball deformation due to the high speed motion of the ball and the movement of the camera, an efficient background subtraction method based on three-frame difference technique is used. This is also useful to extract the moving objects in the foreground in presence of multiple moving objects in the background. A feature based ball-detection framework is used to generate a set of ball candidates. A 2D candidate distribution analysis is used to generate a set of candidate trajectories. For a long shot sequence of basketball video, the ball moves in a parabolic path. Using this physical characteristic of the ball motion, the actual ball trajectory is extracted. The missing ball locations due to occlusion can also be predicted using a simple prediction function. The proposed framework provides a near real-time analysis of the ball trajectory without need of multiple camera set up and complex derivations.

Previous works on ball tracking

Many algorithms have been developed in the recent years to detect and track the ball in the sports video. Seo et al. [13] proposed a Kalman filter based template matching procedure to track a ball in a broadcast soccer video. They also used back projection to predict the occluded ball positions. The main drawback of the algorithm is it requires manual input of the starting locations of the ball and the tracking result was not reported.

D' Orazio et al. [14] used a modified version of directional Circle Hough transform to detect the ball in a soccer game. The algorithm gives poor result when the ball is occluded. The algorithm also fails to identify the ball in the presence of some non-ball objects which look like the ball more than the ball itself in the video.

Yu et al. [15] proposed a trajectory-based ball-detection-and-tracking algorithm in soccer video. They used salient object detection technique to detect the centre circle, goalmouth and the players. The ball candidates are then generated using the shape, size and color properties of the ball. A Kalman filter based prediction system is used to generate the trajectory of the ball. Though this algorithm yields very good result to detect and track the ball in a broadcast soccer video and the computational time was not reported, the complexity of the algorithm seems to be on the higher side.

Chen et al. [16] presented a ball tracking framework based on the 2D trajectory information for broadcast baseball videos. The ball trajectories are generated by analyzing the 2D distribution of the ball candidates. They use *Positive Frame Difference Image* (PFDI) to segment the moving ball in the foreground from the background. But this method tends to fail when the background contains multiple moving objects.

Chen et al. [17] proposed a physics-based ball tracking system to estimate the shooting location in basketball video. The 3D trajectories are reconstructed from the 2D trajectories exploiting the physical characteristics of the ball motion and calibrating the camera parameters. However, the complexity of this algorithm should be high as it requires calculating the camera parameters for 2D to 3D conversion.

Shum and Komura [18] presented a new method to extract and calculate the 3D trajectory of a pitched baseball in a video clip using single-view television clip. The trajectory of the ball is extracted by global search methods of dynamic programming. They used color based segmentation and simple background subtraction to detect the ball in the foreground but this method fails when the ball speed is high and the color of the ball changes considerably with the ball motion.

Liang et al. [19] proposed an algorithm for ball detection and tracking in broadcast soccer video where the ball candidates are generated using color, shape and size features. A weighted graph is constructed using the ball candidates and Viterbi algorithm is used to extract the optimal path as the location of the ball. Kalman filter based template matching is employed to track the ball in subsequent frames. The algorithm fails to detect the ball when the ball is occluded or the ball image is merged with line segments and player's body parts.

Contributions of the proposed framework

The block diagram of the proposed method is shown in Fig. 1. The first step is to segment out the moving objects in every frame which is done using frame differencing method of background subtraction taking three consecutive frames for concurrent processing. The morphological operations are then preformed on the segmented objects to fill the small gaps and remove the noise. An edge detection technique is employed to clearly discriminate the moving objects from the background. The result of the segmentation is a set of ball candidates along with several ball-like objects among which the ball candidates are identified using several filters based on size, shape and compaction feature of the ball image. A 2D candidate distribution plot is generated by plotting the centroid locations of the ball candidates over time, i.e. the number of frames. A set of ball trajectories are formed using a trajectory growing process. The true ball trajectory can be identified by analysing the physical motion and the characteristics of the trajectory of the ball. Trajectory interpolation is done to find the missing ball positions and finally the computed trajectories are superimposed on the original frames to estimate the ball position along the trajectory.

The main contributions of the proposed framework are mentioned below:

1) An integrated framework for automatic detection and tracking of a basketball in a long-shot sequence of basketball sports video is presented. At least two different types of method are used in this framework: feature-based algorithm to generate the set of ball candidates in every frame and data association algorithm based tracking operation to validate the ball locations in the subsequent frames.
2) A robust and fast background subtraction algorithm is used to segment out the moving objects in the foreground in a complex and dynamic scene. Unlike many previous algorithms where simple frame differencing method is used to segment the moving objects in the foreground, the method used in this work yields great results in presence of multiple moving objects in the background. There is also no need to detect and remove the salient objects like border lines, basketball board boarders which in turn reduce the computational time.
3) Use of features like size, shape and compaction minimizes the number of non ball objects in the frame. This reduces the number of ball candidates generated in each frame which are to be processed in the trajectory processing phase and in turn reduces the computational time.
4) The colour feature is omitted deliberately which make it possible to use the framework for detection and tracking of basketball regardless of the colour of the ball used in the game. This also eliminates the chances of wrong detection when the ball colour changes significantly with varying illumination conditions and high speed motion of the ball.
5) In trajectory processing phase, the ball locations in the subsequent frames are predicted using a

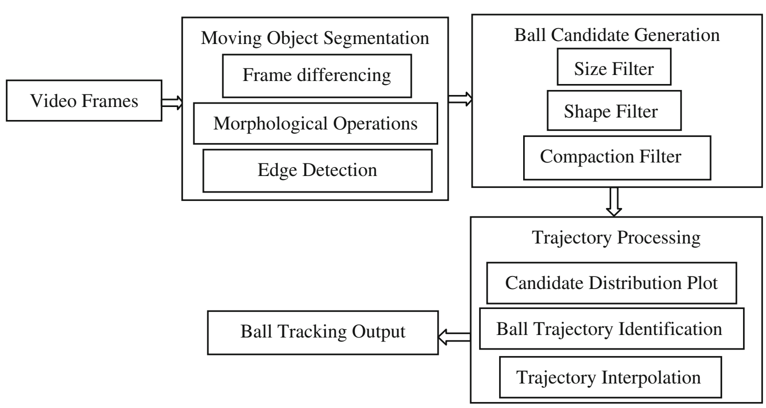


Fig. 1 Block diagram for position estimation and tracking of a basketball from a real time video

simple prediction function and is being updated by the best-fitting technique for the quadratic functions using statistical regression analysis. The physical characteristic of the ball trajectory is used to find out the ball trajectory from a set of trajectories. The missing ball locations are successfully recovered by using interpolation technique.

6) The proposed framework requires input from only one camera, thus eliminating the rigorous calculations required for camera parameters calibration and reduces the computational complexity of the algorithm.

We tested our algorithm with different types of video data. Where some videos are taken manually using a camcorder, a set of video is downloaded from the internet. The proposed framework gives excellent results for both types of video nullifying the effects of different types of noise, e.g. noise generated due to camera movement and shaking and channel noise during transmission. The proposed algorithm yields excellent results for the occluded and merged balls in the video.

We intend to use our algorithm mainly to detect-and-track the basketball during training sessions. Both the amateur and professional players spend a lot of effort to improve their skill during training. While professionals may use hi-tech systems to analyze their performance during training, it is very unlikely for the amateur players to afford a costly set-up for training purpose. We believe our system will give every player and coach the much needed support during training with a highly accurate, low cost and simple set-up. With a few more modifications, we are expecting to use the algorithm for in-depth analysis of any ball game in near future.

Moving object segmentation

Background subtraction

The moving objects present in a video frame are segmented out by using background subtraction method. The background subtraction can be done by using the simple frame differencing method. In this method, the intensity difference of every two consecutive frame is calculated and thresholded to obtain foreground pixels as shown in Eq. (1).

$$Intensity_n(i,j) = \begin{cases} 255, & if\,|Intensity_n(i,j) - Intensity_{n-1}(i,j)| > T_s \\ 0 & Otherwise \end{cases} \tag{1}$$

where n is the number of frames in the video sequence and T_s is the threshold value. In case of fast moving objects like a ball, the *two-frame differencing technique* gives erroneous results as it generates many "ball-like" objects. To incorporate with this, we use a *three-frame differencing technique* [27] where frame difference is performed on three consecutive frames between frames f_n, and f_{n-1} and between frames f_n, and f_{n+1}. The output image of the frame differencing is two difference images d_{n-1}, and d_{n+1} as shown by Eq. (2) and Eq. (3) respectively. A judicially selected threshold T_d is applied on the two difference image to separate the moving object from the background.

$$d_{n-1} = |f_n - f_{n-1}| \tag{2}$$

$$d_{n+1} = |f_n - f_{n+1}| \tag{3}$$

$$d_N(i,j) = \begin{cases} 1, & if\ d_N(i,j) > T_d \\ 0 & Otherwise \end{cases} \tag{4}$$

Where, $N = n\text{-}1$ *and* $n+1$.

An "*AND*" operation is performed on two difference images to find the similarity between the difference images.

$$d = d_{n-1} \cap d_{n+1} \tag{5}$$

The difference calculated using Eq. 5 gives the motion estimation of the moving objects in the foreground, suppressing the background clutter.

Morphological operations

A number of morphological operations [28, 29] are performed on the segmented images mainly to fill the holes created due to the motion of the moving object. The noise due to motion discontinuity of the camera is also greatly reduced by morphological operations. Mainly three types of morphological operations are performed, namely, morphological opening, closing and dilation.

The morphological opening of an input image A by a structuring element B can be mathematically expressed as,

$$A \circ B = \bigcup (B + x : B + x \subseteq A) \tag{6}$$

Morphological opening operation smoothes the object contour and break thin projections due to motion discontinuities.

The morphological closing operation is performed to join narrow breaks and fill holes in the segmented image and can be mathematically expressed as,

$$A \bullet B = (A \oplus B) \Theta B \tag{7}$$

The basic operation of morphological dilation is to gradually enlarge the boundaries of regions of foreground pixels, thus filling up the holes in the boundary. The operation is generally performed on binary images but with little modifications, it can be directly used to gray scale images. The morphological dilation of an input image A with a structuring element B can be given as,

$$A \oplus B = \bigcup \{B + a : a \in A\} \tag{8}$$

Edge detection

The edge detection technique is employed to detect the significant discontinuities in the intensity values in the segmented images. These discontinuities are created by the object motion in the foreground and the camera motion. We use a Canny edge detector [30, 31] to detect the edges because of its superior capability of detecting strong and weak edges together. A Gaussian filter is used to remove the noise and the gradient magnitude and the direction of the edges are calculated as the derivative of the Gaussian filtered image as,

$$|G| = \sqrt{g_x^2 + g_y^2} \quad \text{and} \quad \theta = \tan^{-1}\left(\frac{g_y}{g_x}\right) \tag{9}$$

The local maxima of the gradient magnitude image are marked as edges. A double thresholding is performed to detect the edges. Strong edges are directly included in the final edge image and the weak edges are included if and only if they are connected to strong edges.

The results of moving object segmentation using *three-frame difference* method, morphological opening, closing and dilation operation and edge detection are shown in Fig. 2. It is quite clear from Fig. 2(a) that the three-frame difference method of background subtraction is very effective in discriminating the moving object in the foreground from the objects present in the background. Though there are some noise parts and a portion of backboard border are present after segmentation, these effect can be nullified by morphological operations as shown in Fig. 2(c). It can also be noticed in Fig. 2(d) that, the ball image is not perfectly circular.

Ball candidate generation

Many objects in a frame look like the ball after segmentation. On the other hand, the ball shape gets deformed due to the motion of the ball and the movement of the camera. To filter out the ball candidates from the segmented moving objects, some filters are employed based on the shape, size and compaction features of the ball image. The objects which satisfy the constraints are considered as the ball candidates and retained for further processing.

Size filter

The ball size in a sports video varies in frame to frame because the ball comes closer or moves away from the camera location. The ball size also changes when the ball gets partially occluded by the players or the ball image is merged with other objects in the frame. The segmented moving objects in a frame are filtered out beyond the size limit [$S_{\min}$, $S_{\max}$], where $S_{\min}$ and $S_{\max}$ are the minimum and maximum value of possible ball size for the given video sequence. For a video sequence of frame size 640×480, the observational limit of the size filter lies in the range [5, 27]. Similarly, for video sequence of frame size 480×360, the size filter value lies in the range [4, 9].

Shape filter

In a sports video, the ball is the fastest moving object and so, the shape of the ball varies significantly in almost every frame. In many frames, the ball image does not at all look like a circle as shown in Fig. 2(d) earlier. The presence of camera motion also led to the

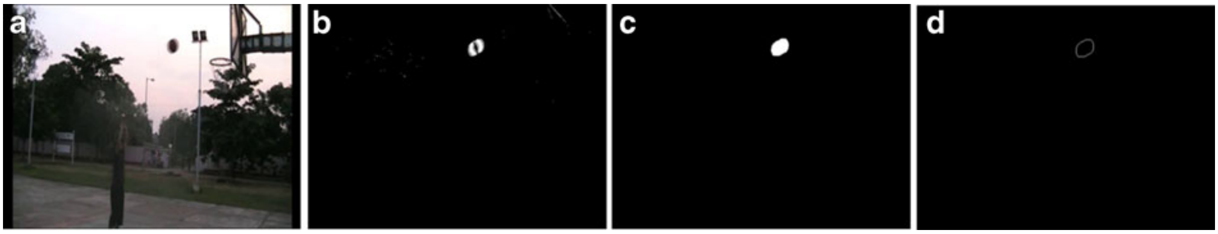


Fig. 2 Example of moving object detection method. **a** Original frame, **b** Moving object segmentation using three-frame difference method, **c** ball image after morphological operation, **d** ball image after applying edge detection method

shape deformation of the ball image. Thus the near-circular ball images should be retained while objects of other shapes should be filtered out to correctly identify the ball in frames. A shape filter is employed based on the aspect ratio of the blobs detected in the frame. A threshold for the shape filter is defined which is set to be 4 for the video data set used in this algorithm. The objects having the aspect ratio within the range [1/4, 4] are retained as probable ball candidates.

Compaction filter

The *compaction* [32] of a given shape is the measure by how much the shape differs from a circle. The maximal possible value of the compaction of a shape is 1, if and only if the given shape is a circle. The degree of compaction C_D is defined as the ratio of square of the perimeter of the given shape to the area of the given shape and is shown mathematically in Eq. (10),

$$C_D = P^2/4\pi A \tag{10}$$

Where P is the perimeter of the object and A is the area of object. Though C_D should be equal to 1 for a ball image, but in practical case, the ball images are far from being a circle in many frames. Thus, a threshold of the compaction filter should be defined. The threshold of compaction filter is to be chosen as 50 %, i.e. the degree of compaction for a ball image should be greater than or equal to half. All objects below the threshold are filtered out as non-ball objects.

Trajectory processing

After the ball candidate generation step, we have a number of ball candidates that includes both true ball candidates and non-ball objects. It is very difficult to identify the ball from the set of ball candidates. On the other hand the ball image lost its shape when it gets occluded with other objects in the frame and thus can be omitted during ball candidate generation process. This eventually leads to a number of *false alarms* in the ball tracking result. To deal with this, a trajectory based detection-and-tracking algorithm is used where a set of trajectories are generated from the set of ball candidates. It is much easier to find the ball using its trajectory information and it also helps to recover the missing ball locations due to occlusion.

Candidate distribution plot

In a long shot sequence of a basketball game, the ball moves in a parabolic path. Hence, a 2D distribution analysis of the ball candidates is used to determine the ball trajectory. A candidate distribution plot is generated by plotting the centroid locations of all the ball candidates over time. The plot represents the location of the ball candidates in each frame. In the basketball video, the ball is the continuous moving object over a number of frames. The ball trajectory can be identified from the candidate distribution plot as one with a smooth and relatively long parabolic trajectory path. The non-ball objects exhibit very short trajectories or no trajectory at all. Figure 3 depicts the 2D candidate distribution plot of the ball candidates.

Ball trajectory identification

The next step is to generate a set of candidate trajectories from the candidate distribution plot and identify the ball trajectory from the set of candidate trajectories. Figure 4 shows the trajectory generation algorithm. At first, the ball candidates in a frame are linked with the nearest neighbours in the next frame.

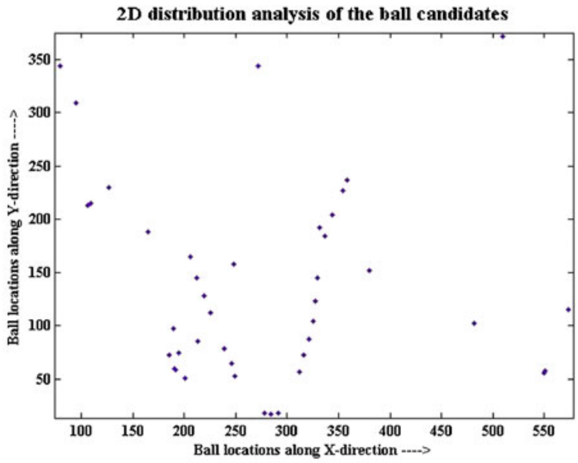


Fig. 3 Candidate distribution plot

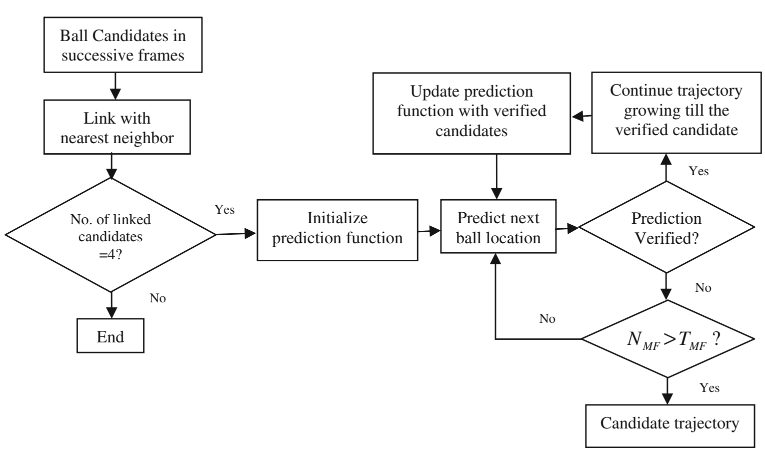
The prediction function used to predict the ball location in the next frame is given by,

$$y = a \bullet n^2 + b \bullet n + c, a > 0 \tag{11}$$

The prediction function is initialized if the number of linked candidate is equal to 4 and the next ball location is predicted. The prediction is said to be "*verified*" if there exists a ball candidate near the predicted location. The trajectory growing procedure is continued by linking the correctly predicted candidates and the prediction function is updated using the best-fitting function for the candidates. The prediction error is calculated as the vertical distance from the location of the ball candidate and the parabolic curve. A threshold of prediction error is defined to correctly identify the ball locations. The best-fitting quadratic function for each ball location is calculated using the statistical regression analysis. In case the predicted location does not match with the original location of the ball candidate in a frame, the prediction function continues to predict the locations for the next few frames until the prediction is verified by a ball candidate. The trajectory is then extended to the verified frame and the predicted locations are considered as the ball positions in the intermediate frames. The intermediate frames that are not verified are termed as "*missing frames*". The trajectory growing procedure terminates when the number of consecutive missing frames N_{MF} exceeds a threshold value (T_{MF}). The algorithm then selects a new pair of ball candidates in successive frames and repeats the same procedure as shown in Fig. 4.

To identify the ball trajectory from the set of candidate trajectories, the physical characteristics of ball motion over the frame is considered. For a long shot sequence in a basketball game, the ball trajectory is a parabolic curve due to the gravity of the earth. Furthermore, in a basketball video sequence, the ball is the continuously moving object in the frames. Thus, the ball trajectory should be the longest among set of candidate trajectories generated by other moving objects in the frame. The length of the trajectory can be calculated as the number of consecutive ball candidates linked together to form the trajectory. To extract the ball trajectory from the set of candidate trajectories, the *prediction error* is also considered as a criterion. The *prediction error* is calculated as the average distance from each predicted ball location to the original ball candidate position. A threshold of the prediction error is defined with sufficient error tolerance. The candidate trajectories having high values of prediction error are

Fig. 4 Flowchart of the candidate trajectory generation algorithm



eliminated. Thus, the ball trajectory can be identified based on the trajectory length and shape and the prediction error. Figure 5 shows the ball trajectory identification procedure and the statistical regression analysis method to obtain the ball position in the video.

As we can see from Fig. 5(a), a set of candidate trajectories are generated by linking the ball candidates. Furthermore, the true ball trajectory is identified based on the physical characteristics of ball motion, i.e. the ball trajectory for a long shot sequence should form a parabolic curve and the length of the ball trajectory should be longest among all the candidate trajectories generated. A 95 % prediction bound is used for the predicting the locations of the ball along the trajectory.

Trajectory interpolation

In a basketball game, the ball often gets occluded with the player's body part. The ball images are also get merged with the backboard borders and other objects in the background. In many cases the ball-detection-and-tracking algorithm fails to detect the ball due to the deformation of the ball images. To recover the missing ball positions, a trajectory interpolation method is used. The flowchart of the trajectory interpolation algorithm is presented in Fig. 6.

The ball location along a trajectory is predicted using Eq. 11 and the prediction is verified with the location of ball candidate in that frame. If there exists a ball candidate that satisfies the threshold value T_{MF} of the prediction, the trajectory growing algorithm continues to extend the trajectory to the next frame. If the verification fails, the prediction of the ball location is continued up to a few frames, i.e. up to *$(n + T_{MF})$-th* frame by updating the variables of the prediction function. Let, prediction is continued up to *k-th* frame where the value of k is less than or equal to T_{MF}. If the predicted location in *k-th* frame is verified by a ball candidate, the trajectory is extended up to *$(n + k)$-th* frame. The intermediate frames are called *missing frames* and the predicted locations are considered as the ball location in that frames. Once the missing ball positions are correctly estimated, the ball locations are plotted and superimposed on the original frame to verify with actual ball flight path.

Experimental results and discussions

Performance analysis of the proposed framework

The experiments were performed in MATLAB platforms using two different video data sets. Each data base contains four different videos of basketball long shot sequences. The first data base, BS, contains videos taken using a Canon camcorder at the outdoor basketball court at our institution. The second database DS contains videos that are downloaded from various internet sources.

The ball position in each video sequence is obtained by manual inspection and the ball location in each frame is recorded. The experimental findings of the ball detection-and-tracking solution for BS sequences are summarized in Table 1, where "*Ball detection results*" gives the results of the ball detection using different features of the ball image. The "*Ball*

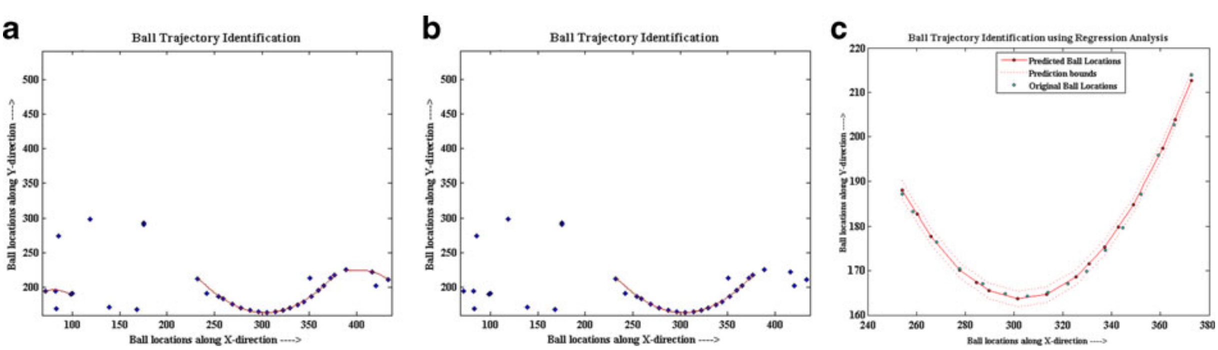


Fig. 5 Candidate trajectory generation and ball trajectory identification using the regression analysis method. **a** Set of candidate trajectories generated from the candidate distribution plot. **b** The identified ball trajectory. **c** The method of regression analysis. The predicted ball locations are shown by *red dots* and the original position of the ball candidates are shown by *green dots*. The *dashed line* depicts the bound of the prediction error

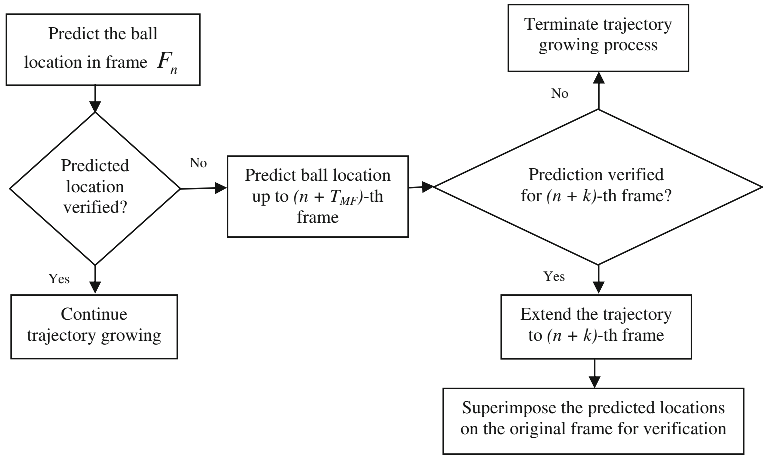


Fig. 6 Flowchart of the trajectory interpolation algorithm

tracking results" gives the final output after the trajectory processing. The row "*correct*" represents the number of frames where the algorithms detects a ball in a ball frame in its actual location and does not detect any ball in a no-ball frame. "*False*" represents the number of frames where a ball is detected at an incorrect location in a ball frame or a ball is detected in a no-ball frame. It also includes the frames where the ball is not detected in a ball frame. "*Tracked*" represents the number of frames where the predicted ball location is matched with the actual ball location in that frame and "*miss*" represents the case where the predicted ball location does not match with the actual ball location in a frame.

In BS sequence, there are total 344 frames containing both ball frames and no-ball frames. From Table 1 it is quite evident that the ball detection algorithm works efficiently in presence of multiple moving objects in the background and attains an average accuracy of 96.22 %. The rate of misses is also as low as 3.78 %. The misses are mainly because of the occlusion of the ball with other objects in the frame and the deformation of the ball image due to the motion blurring. The ball tracking results gives an overwhelming precision of 100 % for all the four video sequences where the ball location in each frame is accurately predicted along with the missing ball locations.

Next, the proposed algorithm is tested with four basketball video sequences of the DS video. The experimental results are summarized in Table 2. The task is more difficult for the DS video because of the different acquisition conditions and the presence of transmission noise in some cases. Still the proposed ball detection algorithm attains an overall accuracy of 92.52 % for a total number of 254 frames. Especially, in DS-001 and DS-002 video sequences, the accuracy of ball detection is a bit low, that is 88.33 % and 92.7 % respectively. But again the trajectory processing algorithm works accurately for all four videos and we get the precision as 100 % along with the interpolated ball locations.

The experimental results for ball detection-and-tracking for video data set BS and DS are shown in Fig. 7 and Fig. 8 respectively. It can be noticed that the

Table 1 Performance analysis of the proposed algorithm for BS sequence

Video sequence	Ball detection results				Ball tracking results			
	Total frames	Correct	False	Accuracy (%)	Ball frames	Tracked	Miss	Precision (%)
BS-001	90	86	04	95.5	20	20	00	100
BS-002	50	46	04	92.0	25	25	00	100
BS-003	64	62	02	96.87	19	19	00	100
BS-004	140	137	03	97.85	29	29	00	100
Total	344	331	13	96.22	93	93	00	100

Table 2 Performance analysis of the proposed algorithm for DS sequence

Video sequence	Ball detection results				Ball tracking results			
	Total frames	Correct	False	Accuracy (%)	Ball frames	Tracked	Miss	Precision (%)
DS-001	60	53	07	88.33	30	30	00	100
DS-002	69	64	05	92.70	31	31	00	100
DS-003	62	58	04	93.54	33	33	00	100
DS-004	63	60	03	95.23	25	25	00	100
Total	254	235	19	92.52	119	119	00	100

detection error occurs mostly because of the merging of the ball image with trees, lines and borders in the field and the deformation of the ball image due to motion blurring. For example, in Fig. 7(a), the algorithm fails to detect the ball where it gets merged with trees in the background. It can also be noticed that the effect of the motion blurring is much higher at the points where the ball is released from the player's hand. The missing ball locations are successfully predicted along the ball trajectory as shown in Fig. 7(d). The proposed algorithm works efficiently for both video data sets irrespective of the different acquisition condition that includes different types of cameras and background.

The experiments were performed on a HP desktop system (CPU: Intel Core i5, 2.4 GHz, RAM: 3 GB). For a long shot sequence of 4 s, the processing time is about 10–11 s, that is, the proposed framework gives the basketball tracking results almost in real-time basis. This can be very helpful for training purpose to determine the throwing angle of the ball from any location at the basketball court. It can also be used in broadcast basketball game videos to classify the shot sequences and determine the shooting location in a real-time basis.

For performance comparison we have implemented a Kalman filter based ball detection algorithm. Kalman filter [33, 34] is extensively used for object

Fig. 7 Position estimation of basketball for BS video sequences with missing ball position estimation. The detected ball locations are shown by *green dots* and the interpolated ball positions are shown by *yellow dots*. **a**, **b**, **c**: Estimation of the ball locations in different video sequences. **d**, **e** and **f**: Interpolated ball locations to recover the missing ball positions along the ball trajectory

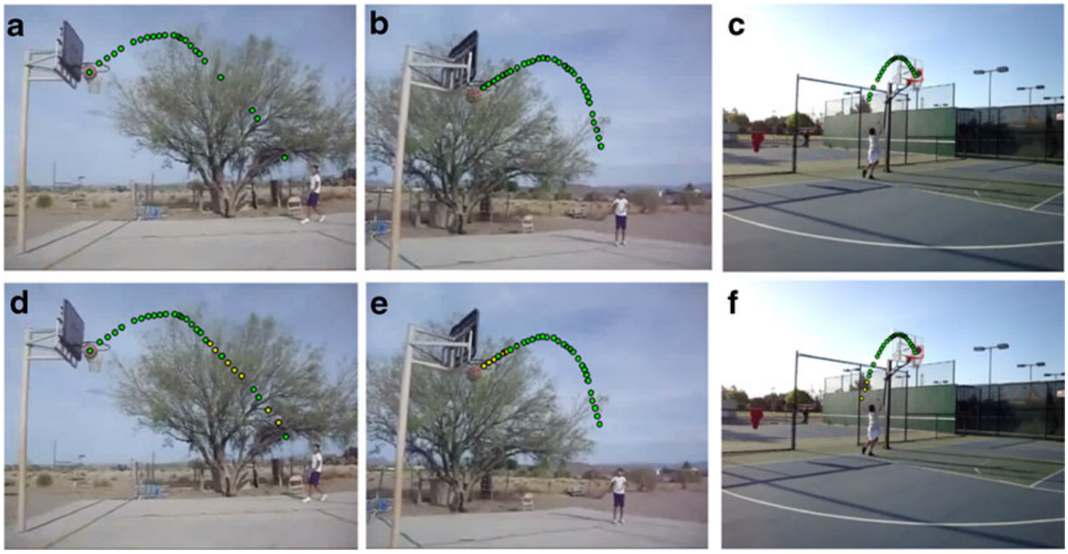


Fig. 8 Position estimation of basketball for DS video sequences with missing ball position estimation. The detected ball locations are shown by *green dots* and the interpolated ball positions are shown by *yellow dots*. **a**, **b**, **c**: Estimation of the ball locations in different video sequences. **d**, **e** and **f**: Interpolated ball locations to recover the missing ball positions along the ball trajectory

tracking and gives very good results with its inbuilt capacity of predicting the locations of a moving object. For the time being we are considering the ball detection result of our algorithm for comparison without applying the trajectory processing phase. We use the accuracy, precision of ball detection and number of ball candidates generated per video as criteria. The comparison results are shown in Table 3. Here "*accuracy*" refers to detecting a ball at its original position in a ball frame and not detecting a ball in a no-ball frame. The term "*precision*" indicates the number of frames where the ball is detected among all the ball frames in a particular video sequence.

As it can be noticed in Table 3, the accuracy of ball detection for the proposed algorithm using different features of ball image is much higher than the Kalman filter based direct ball detection method. The average accuracy of ball detection for the proposed method is 94.65 % where that of Kalman filter based method is 84.95 %. The precision of ball detection for the proposed method is also much higher (89.62 %) than the Kalman filter based method (71.7 %). Another

Table 3 Comparison of the proposed algorithm and Kalman filter based ball detection method for BS and DS video sequences

Video sequence	Proposed framework			Kalman filter based method		
	Accuracy (%)	Precision (%)	# of ball candidates generated	Accuracy (%)	Precision (%)	# of ball candidates generated
BS-001	95.5	80.0	22	91.1	75.0	36
BS-002	92.0	84.0	36	82.0	76.0	38
BS-003	96.87	89.47	17	82.81	57.89	69
BS-004	97.85	89.65	45	88.57	62.07	118
DS-001	88.33	76.67	33	71.67	60.0	121
DS-002	92.7	83.87	57	81.59	64.51	173
DS-003	93.54	87.88	60	91.93	93.9	84
DS-004	95.23	88.0	41	88.89	80.0	102

important advantage of the proposed algorithm is it generates less number of ball candidates than the Kalman filter based method. That means, the proposed method is more effective and it also helps to reduce the complexity and processing speed of the trajectory processing block. Figure 9 gives a pictorial reference of the comparison analysis for the proposed method and Kalman filter based method. It is very clear from Fig. 9(a), that the proposed method gives much better results in terms of accuracy of ball detection. Among the eight video sequences used for simulation, in only one case (for video sequence *DS-003*) the Kalman filter based method gives superior result than the proposed method as shown in Fig. 9(b).

The proposed algorithm is simple and easy to implement. Unlike using the ball velocity constraints to perform the ball tracking used by Chen et.al [17], a much simpler method is used by exploiting the physical characteristics of the ball motion. The actual ball trajectory can be easily identified from the set of candidate trajectories using the physical characteristics of the ball motion and comparing the length of the candidate trajectories. Though the effects of air friction and ball-spin are not taken under consideration, still the proposed framework gives excellent results for outdoor environment.

Conclusion

In this paper, a framework is developed to detect and track a basketball in a video of basketball long shot sequence. The algorithm is designed based on two key ideas. The first idea is to generate a set of ball candidates per frame. Unlike the direct detection method, it greatly reduces the number of misses because the deformed ball images due to occlusion, merging and motion blurring are also included in the set of ball candidates. The moving objects in the foreground are segmented from the moving objects in the background by using a three-frame difference method. This in turn reduces the effect of motion blurring. The ball candidates are filtered out based on the size, shape and compaction constraints. The second idea is to use the ball trajectory for more efficient tracking. A set of candidate trajectories are generated from the 2D distribution analysis of the ball candidates over time. The actual ball trajectory is identified using the physical characteristics of ball motion during a long shot sequence and the length of the trajectory. In a long shot sequence of a basketball game, the ball moves in a near parabolic path and the length of the ball trajectory is generally longest among all the other trajectories because the ball is the continuously moving object in subsequent frames. The missing ball locations can also be predicted by an interpolation technique. The final ball locations are superimposed along the ball trajectory in the video which shows the ball flight path and can be used for extensive analysis of the shot.

The future work includes making the existing algorithm more robust by introducing a motion de-blurring module. With a few modifications in the algorithm, a system can be developed to detect and track the ball and the players separately in the video. Currently we are exploring to estimate the shooting angle of the ball using the ball velocity constraints along with the shooting location of the player. We have used the same approach for volleyball game successfully and used the results of ball tracking to determine the set-type in a particular video [35]. We are planning to apply this approach for other detection-and-tracking problems such as traffic monitoring and implement various surveillance tracking systems.

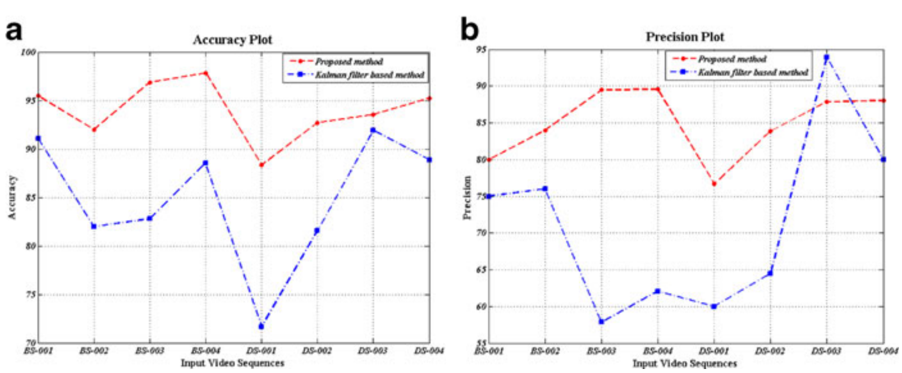


Fig. 9 Comparison of the proposed method with Kalman filter based method. The *red dashed line* shows the values of accuracy and precision for the proposed method and the *blue dashed line* shows the accuracy and precision for the Kalman filter based method. **a** Accuracy plot and **b** Precision plot

References

1. Y. Rui, A. Gupta, A. Acero, Automatically extracting highlights for TV baseball programs, in *Proc. of 8th International Conference on Multimedia* (2000), pp. 105–115
2. J. Assfalg, M. Bertini, C. Colombo, D. del Bimbo, W. Nunziati, Semantic annotation of soccer videos: automatic highlights detection. Comp. Vision Image Underst. **92**, 285–305 (2003)
3. D. Yow, B.L. Yeo, M. Yeung, B. Liu, Analysis and presentation of soccer highlights from digital video, in *Proc. of 2nd Asian Conference on Computer Vision (ACVV'95)* (1995), pp. 499–503
4. S. Nepal, U. Srinivasan, G. Reynolds, Automatic detection of 'goal' segments in basketball videos, in *Proc. of the 9th ACM International Conference on Multimedia* (2001), pp. 261 doi:10.1145/500141.500181
5. B. Li, I. Sezan, Event detection and summarization in sports video, in *IEEE Workshop on Content-Based Access of Image and Video Libraries (CBAIVL'2001)* (2001), pp. 132–138 doi:10.1109/IVL.2001.990867
6. X. Yu, D. Farin, Current and emerging topics in sports video processing, in *IEEE International Conference on Multimedia and Expo* (2005), pp. 526–529 doi:10.1109/ICME.2005.1521476
7. J.R. Wang, N. Parameswaran, Survey of sports video analysis: research issues and applications, in *Proc. Pan-Sydney Area Workshop on Visual Information and Processing (VIP'2003)* (2003), pp. 87–90
8. G. Sudhir, J.C.M. Lee, A.K. Jain, Automatic classification of tennis video for high-level content-based retrieval, in *IEEE International Workshop on Content-Based Access of Image and Video Database* (1998), pp. 81–90 doi:10.1109/CAIVD.1998.646036
9. J.R. Wang, N. Parameswaran, Detecting tactics patterns for archiving tennis video clips, in *Proc. of IEEE 6th International Symposium on Multimedia Software Engineering* (2004), pp. 186–192
10. T. Taki, J. Hasegawa, T. Fukumura, Development of motion analysis system for quantitative evaluation of teamwork in soccer games, in *Proc. of International Conference on Image Processing*, vol. 3 (1996), pp. 815–818, doi:10.1109/ICIP.1996.560865
11. J. Liu, X. Tong, W. Li, T. Wang, Y. Zhang, H. Wang, Automatic player detection, labelling and tracking in broadcast soccer video. Pattern Recognit. Lett. **30**(2), 103–113 (2009)
12. J. Xing, H. Ai, L. Liu, S. Lao, Multiple player tracking in sports video: a dual-mode two-way Bayesian inference approach with progressive observation modeling. IEEE Trans. Image Process. **20**, 1652–1667 (2011)
13. Y. Seo, S. Choi, H. Kim, K. Hong, Where are the ball and players? Soccer game analysis with color based tracking and image mosaic, in *Proc. ICIAP* (1997), pp. 196–203
14. T. D'Orazio, N. Ancona, G. Cicirelli, M. Nitti, A ball detection algorithm for real soccer image sequences, in *Proc. of 16th International Conference on Pattern Recognition*, vol. 1 (2002), pp. 210–213 doi:10.1109/ICPR.2002.1044654
15. X. Yu, H.W. Leong, C. Xu, Q. Tian, Trajectory-based ball detection and tracking in broadcast soccer video. IEEE Trans. Multimed. **8**(6), 1164–1178 (2006)
16. H.T. Chen, H.S. Chen, M.H. Hsiao, W.J. Tsai, S.Y. Lee, A trajectory-based ball tracking framework with visual enrichment for broadcast baseball videos. J. Inf. Sci. Eng. **24**, 143–157 (2008)
17. H.T. Chen, M.C. Tien, Y.W. Chen, W.J. Tsai, S.Y. Lee, Physics-based ball tracking and 3D trajectory reconstruction with applications to shooting location estimation in basketball video. Elsevier Sci. J. Vis. Commun. Image Represent. **20**, 204–216 (2009)
18. H. Shum, T. Komura, A spatiotemporal approach to extract the 3D trajectory of the baseball from a single view video sequence, in *IEEE International Conference on Multimedia and Expo* (2004), pp. 1583–1586
19. D. Liang, Y. Liu, Q. Huang, W. Gao, *A scheme for ball detection and tracking in broadcast soccer video, vol. 3767* (Springer, Berlin, 2005), pp. 864–875
20. W. Chen, Y.J. Zhang, Tracking ball and players with applications to highlight ranking of broadcasting table tennis video, in *Proc. of IMACS Multiconference on Computational Engineering in Systems Applications* (2006), pp. 1896–1903
21. K. Choi, B. Park, S. Lee, O. Seo, Tracking the ball and players from multiple football videos. Int. J. Inf. Acquis., World Scientific Publishing Company, 1–8 (2006)
22. Hawk-Eye, http://www.hawkeyeinnovations.co.uk
23. S. Hashimoto, S. Ozawa, A system for automatics judgment of offsides in soccer games, in *IEEE International Conference on Multimedia and Expo* (2006), pp. 1889–1892 doi:10.1109/ICME.2006.262924
24. L.Y. Duan, M. Xu, Q. Tian, C.S. Xu, J.S. Jin, A unified framework for semantic shot classification in sports video. IEEE Trans. Multimed. **7**, 1066–1083 (2005). doi:10.1109/TMM.2005.858395
25. W. Hua, M. Han, Y. Gong, Baseball scene classification using multimedia features. Proc. IEEE Int. Conf. Multimed. Expo **1**, 821–824 (2002). doi:10.1109/ICME.2002.1035908
26. K. Wan, X. Yan, X. Yu, C. Xu, Robust goal-mouth detection for virtual content insertion, in *Proc. of the 11th ACM International Conference on Multimedia* (2003), pp. 468–469 doi:10.1145/957013.957116
27. B. Sugandi, H.S. Kim, J.K. Tan, S. Ishikawa, Tracking of moving object by using low resolution images, in *Proc. of International Conference on Innovative Computing, Information and Control* (2007), pp. 408–411
28. R.M. Haralick, S.R. Sternberg, X. Zhuang, Image analysis using mathematical morphology. IEEE Trans. Pattern Anal. Mach. Intell. PAMI- **9**(4), 532–550 (1987). doi:10.1109/TPAMI.1987.4767941
29. P.J. Soille, *Morphological Image Analysis: Principles and Applications*, 2nd edn. (Springer, New York, 2003)
30. J. Canny, A computational approach to edge detection. IEEE Trans. Pattern Anal. Mach. Intell. PAMI- **8**(6), 679–698 (1986)
31. R.C. Gonzalez, R.E. Woods, *Digital Image Processing*, 3rd edn. (Pearson Education, 2008), pp. 719–723
32. R.S. Monetero, State of the art of compactness and circularity measures. Int. Math. Forum **4**, 1305–1335 (2009)

33. N.M. Namazi, P. Penafiel, C.M. Fan, Nonuniform image motion estimation using Kalman filtering. IEEE Trans. Image Process. **3**(5), 678–683 (1994)
34. G. Welch, G. Bishop, An introduction to Kalman filter, in *University of North Carolina at Chapel Hill, NA* (2006), pp. 1–16, www.cs.unc.edu/~welch/media/pdf/kalman_intro.pdf
35. B. Chakraborty, S. Meher, A trajectory-based ball detection and tracking system with application to shot-type identification in volleyball videos, in *Proc. of 9th IEEE International Conference on Signal Processing and Communications (SPCOM 2012)* (Indian Institute of Science, Bangalore, 2012), pp. 1–5, doi:10.1109/SPCOM.2012.6290005