J. Vis. Commun. Image R. 24 (2013) 1191–1203

Contents lists available at ScienceDirect

J. Vis. Commun. Image R.

journal homepage: www.elsevier.com/locate/jvci

Recognizing jump patterns with physics-based validation in human moving trajectory

Hua-Tsung Chen [a,*], Kuo-Lian Ma [b], Jen-Hui Chuang [b], Horng-Horng Lin [c]

[a] Information and Communications Technology Lab, National Chiao Tung University, Hsinchu 300, Taiwan
[b] Department of Computer Science, National Chiao Tung University, Hsinchu 300, Taiwan
[c] Department of Computer Science and Information Engineering, Southern Taiwan University, Tainan 710, Taiwan

ARTICLE INFO

Article history:
Received 17 August 2012
Available online 20 August 2013

Keywords:
Sports video analysis
Camera calibration
Content-based multimedia analysis
3D trajectory approximation
Object tracking
Pattern recognition
Volleyball video
Trajectory analysis

ABSTRACT

This paper presents an approach to recognize jump patterns in human moving trajectory, differentiating jump tracks from planar moving tracks. Since human moving trajectory is one of the most informative representations for content understanding and event detection, trajectory-based video analysis has been gaining popularity. However, a jump action typically leads to violent change in human moving trajectory, since the person suddenly leaves the original plane on which he/she has been moving. The abnormal tracks of the trajectory would influence the performance of trajectory-based video analysis. Hence, differentiating jump tracks from planar moving tracks is of vital importance, not to mention that jump actions typically imply significant events, especially in sports games. In this paper, volleyball videos are used as case study to demonstrate the effectiveness of our proposed jump pattern recognition approach. We derive player trajectory by head tracking, analyze the movement of each player, and recognize potential jump tracks in player trajectories based on two important characteristics: (1) jumps cause pulse-like tracks in the trajectory and (2) the extensions of such tracks go through the vanishing point of vertical lines in the scenes. Finally, the jump positions/heights are estimated, in addition to the planar moving trajectory of each player on the court ground. The experiments show that satisfactory results can be obtained with the proposed recognition scheme.

1. Introduction

1.1. Motivation

The rapid advance in video production technology inspires manifold research issues, including content-based media analysis [1,2], copyright protection [3,4], video coding [5], etc. Especially, the proliferation of digital videos necessitates the development of automatic systems and tools for sematic video content understanding, analysis, and retrieval. Most of the traditional approaches rely on low-level features. However, humans interpret video in terms of semantics rather than low-level features. The demand for automatic video understanding and interpretation requires the mid-level representations mapping from low-level features to high-level semantics, such as shot class, camera motion pattern, color layout, object shape and object trajectory [1,2,6,7]. Object trajectory is one of the most informative representations which are frequently used by humans to analyze events. Hence, trajectory-based video analysis [8–14] has been gaining popularity. In human moving trajectory, a jump action typically leads to a violent change since the person suddenly leaves the original plane on which he/she has been moving. The abnormal tracks of the trajectory are likely to influence the trajectory-based video analysis. Hence, differentiating jump tracks from planar moving tracks is of vital importance. Furthermore, jump actions typically imply significant events, especially in sports games. For example, jump is related to *attack*—the most effective way to score in volleyball. As an important multimedia content, sports video has attracted considerable research efforts due to commercial benefits, and demands of entertaining functionality from the audience [6,7,12–24]. While some approaches of event detection and tactic analysis in sports video have been developed based on player trajectory, the situation of player jumping has rarely been considered in the literature. Hence, we are motivated to recognize jump patterns in human moving trajectory so as to differentiate jump tracks from planar moving tracks.

1.2. Related works

Object trajectory is one of the most informative mid-level representations which can bridge the semantic gap between low-level features and high-level events. Piciarelli et al. [8] propose a trajectory clustering method for video surveillance and monitoring sys-

* Corresponding author. Fax: +886 3 573 6491.
E-mail address: huatsung@cs.nctu.edu.tw (H.-T. Chen).

1047-3203/$ - see front matter © 2013 Elsevier Inc. All rights reserved.
http://dx.doi.org/10.1016/j.jvcir.2013.08.006

tems wherein the clusters are dynamic and built in real-time. The obtained trajectory clusters can provide proper feedback to the low-level tracking system and collect valuable information for the high-level event analysis modules. Su et al. [9] link the local motion vectors across consecutive video frames to form "motion flows," which are recorded and stored in a video database. For video retrieval, noise motions are filtered out and the retrieval process is triggered by query-by-sketch or query-by-example. Assuming trajectory information is already available, Bashir et al. [10] present an efficient motion trajectory-based indexing and retrieval mechanism for video sequences, aiming at solving the problem of trajectory representation when only partial trajectory information is available due to occlusion. Hsieh et al. [11] propose a hybrid motion-based video retrieval system through trajectory matching. First, a curve fitting technique is used by a sketch-based method to interpolate some missing data for the associated control points so that the visual distance between a pair of trajectories can be measured. Then, a string-based scheme is adopted to compare the two trajectories according to their syntactic meanings. With the help of the syntactic distance, a large number of inappropriate candidates can be filtered out and the accuracy of video retrieval can be greatly enhanced.

There has been an explosive growth in the research area of sports video analysis due to the large audience base and the tremendous commercial value. Zhu et al. [12] analyze the temporal–spatial interaction among the ball and the players to construct a tactic representation, *aggregate trajectory*, based on multiple trajectories in soccer video. The tactical patterns are analyzed using the tactic representations which include play region and aggregate trajectory. Yu et al. [13,14] present a trajectory-based algorithm for ball detection and tracking in soccer video. The ball size is first estimated from reference objects (goalmouth and ellipse) so that sequences of ball candidates can be detected and connected into potential trajectories. Finally, the true trajectory is extracted from these potential trajectories by a Kalman filter-based verification procedure. Analyzing tennis video, Han et al. [15] detect the court net and court lines for camera calibration. Players are tracked by the mean-shift method with their real-world positions being used to classify events of service, net approach, and baseline rally. Based on the camera modeling of [15], Han et al. further propose a mixed-reality system in [16]. By changing the parameters of the original camera, a variety of mixed-reality scenes can be synthesized for scene visualization on mobile devices. Luo et al. [17] interpret and analyze human motion in sports video using video object extraction, semantic event modeling, and the Dynamic Bayesian Network (DBN) for characterizing the spatial–temporal nature of the semantic objects. Zhu et al. [18,19] recognize the player actions by considering the movement of body parts for semantic and tactic analysis in tennis video. The *affective* features which simulate a user's emotion are extracted from player actions and trajectories for highlight ranking. Our previous works [20–22] perform physics-based ball tracking in sports video to provide trajectory-based game analysis, such as pitch evaluation in baseball, set type recognition in volleyball, and shooting location estimation in basketball.

To meet the sports-professional's requirement, Hu et al. [23] propose a robust camera calibration method for broadcast basketball video, which extracts player trajectories by a CamShift-based tracking method and maps player trajectories to the real-world court model. The player position/trajectory information is further utilized for professional-oriented applications, including wide-open event detection, trajectory-based target clips retrieval, and tactic inference. However, they do not mention the case of player jumping, which often happens in basketball. Thomas et al. [24] present a particle filter-based approach to track players in beach volleyball using a single camera. With camera calibration, the player trajectory can be mapped to real-world court plane. However, the players are off the ground plane during jumps, resulting in incorrect estimation of the real-world player positions. Fig. 1 gives a sample result of [24]. One can see that a jump leads to an abnormal track in the trajectory, which may be misinterpreted as a movement of the player across the center line of the court.

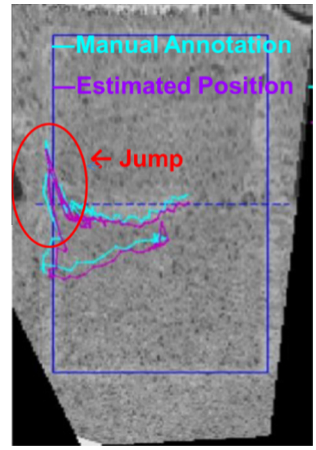

Fig. 1. Player tracking result of [24].

1.3. System framework and contribution

With regard to the foregoing limitations of existing works, we propose a video analysis system for jump pattern recognition in this paper. Fig. 2 illustrates the flowchart of the proposed framework, which contains three main components: camera calibration, head tracking, and the core module—jump pattern recognition. Utilizing a set of corresponding points, the camera calibration aims to compute the geometric transformation between image positions and real-world coordinates. Assuming that the information of each player's height is available, the "ground" trajectories of players over frames are derived by mapping the tracked head positions onto the court plane with the above transformation. Then, the jump patterns in these trajectories are recognized, with the associated jump points/heights estimated. Since this paper mainly focuses on the approach of jump pattern recognition and validates the effectiveness of the proposed method, we choose less complex test sequences of 2-on-2 volleyball games where players rarely occlude one another.

The main contributions of our work are summarized as follows. Jump is an important and frequent action, especially in sports games, but it is rarely mentioned and considered in the literature. Hence, we put forward a jump pattern recognition framework based on some important characteristics. First, it is well known that the extensions of vertical lines in the video frame intersect at a vanishing point. In this paper, we further identify that extensions of pulse-like tracks caused by jumps in a player's moving trajectory will also go through the same vanishing point. To the best of our knowledge, this characteristic has never been used or even been mentioned in the literature. Thus, an effective jump track detection approach based on this newly observed characteristic is designed. Furthermore, considering the theorem of gravitational acceleration and Newton's Second Law of Motion, a physics-based approach is proposed to validate the detected jump tracks for possible reduction of false alarms. In general, jump pattern recognition can be extensively applied to many domains to detect jump-related events, such as attack in volleyball, dunk in basketball, etc. On the other hand, the ability of differentiating the jump tracks

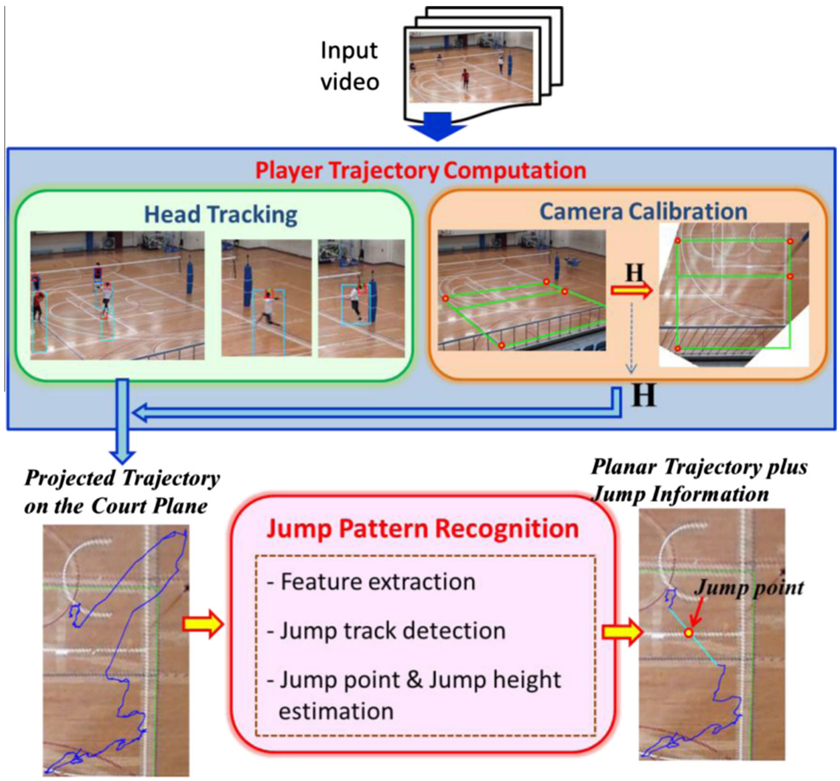

Fig. 2. Flowchart of the proposed jump pattern recognition framework.

and the planar ones will greatly assist the existing as well as future research on trajectory-based video analysis.

The rest of the paper is organized as follows. Section 2 describes player trajectory computation, including head tracking and camera calibration. Section 3 elaborates the processing steps of jump pattern recognition and physics-based validation. In Section 4, experimental results are reported and discussed. Finally, Section 5 concludes this paper. Note that in the following explanation of each processing module, we use the word "trajectory" to designate the linked positions of a moving object (a person), to show the complete movement, and the word "track" to designate a part of a trajectory. Jump tracks indicate the parts of a trajectory where the person jumps, and moving tracks are for the remaining planar movements.

2. Player trajectory computation

In this section, we produce player trajectories via head tracking wherein the skin and hair colors are used as features for head position adjustment. The trajectories are then mapped onto the real-world court plane by homographic transformation so as to facilitate trajectory-based event detection.

2.1. Head tracking

Owing to its effectiveness, the background subtraction method [25] is adopted for foreground object segmentation in our work, with the processing steps shown in Fig. 3. The background model, original frame and segmented foreground objects are presented in Fig. 3(a)–(c), respectively. In most cases, it is reasonable to assume that the top point of each foreground object is the player's head position. However, a player's hand might be higher than his/her head when he/she makes a spike or tosses the ball. Hence, we include another feature—"color" for head detection. We compute the distribution of "hair color" from numerous training head images, which are pre-segmented manually, and set the range of hair color. The foreground pixels of which the colors are not within the hair-color range are discarded. Then, we can take the top hair-colored point of each foreground object as the head position. Fig. 3(d) shows the obtained hair-colored pixels from Fig. 3(c), and (e) presents the result of head detection, where red rectangles indicate the detected head positions.

The head position may not be detected accurately in some cases, such as when a player is occluded by another player or if a foreground object is not segmented correctly. To cope with such problems, head positions in consecutive frames can be checked. Consider the limitations of human kinematics. If the distance of the head positions between two consecutive frames is greater than a threshold, there must be some errors in the detected head positions and certain adjustment or correction process is required.

For head position adjustment, we use a 9×9 search window to find the location near the head position in the previous frame such that the window contains the most skin- and hair-colored pixels. The range of skin color is set in the way similar to that for the hair color. The setting for the hair and skin color ranges will be explained in detail in the experimental section. An example is shown in Fig. 4. Fig. 4(a) and (b) are two consecutive video frames, and the current head position of the player indicated by an arrow in Fig. 4(b) is mis-detected due to the incorrect player segmentation. The red point in Fig. 4(c) is the player's head position in the previous frame and the green rectangle indicates the location of the search window containing the most skin- and hair-colored pixels (presented in white). The green point which is the top point of the search window is then taken as the adjusted head position. Fig. 4(d) shows that reasonable head position can thus be obtained.

2.2. Camera calibration

Camera calibration is an essential task to establish geometric transformations for mapping the positions of the players in the vi-

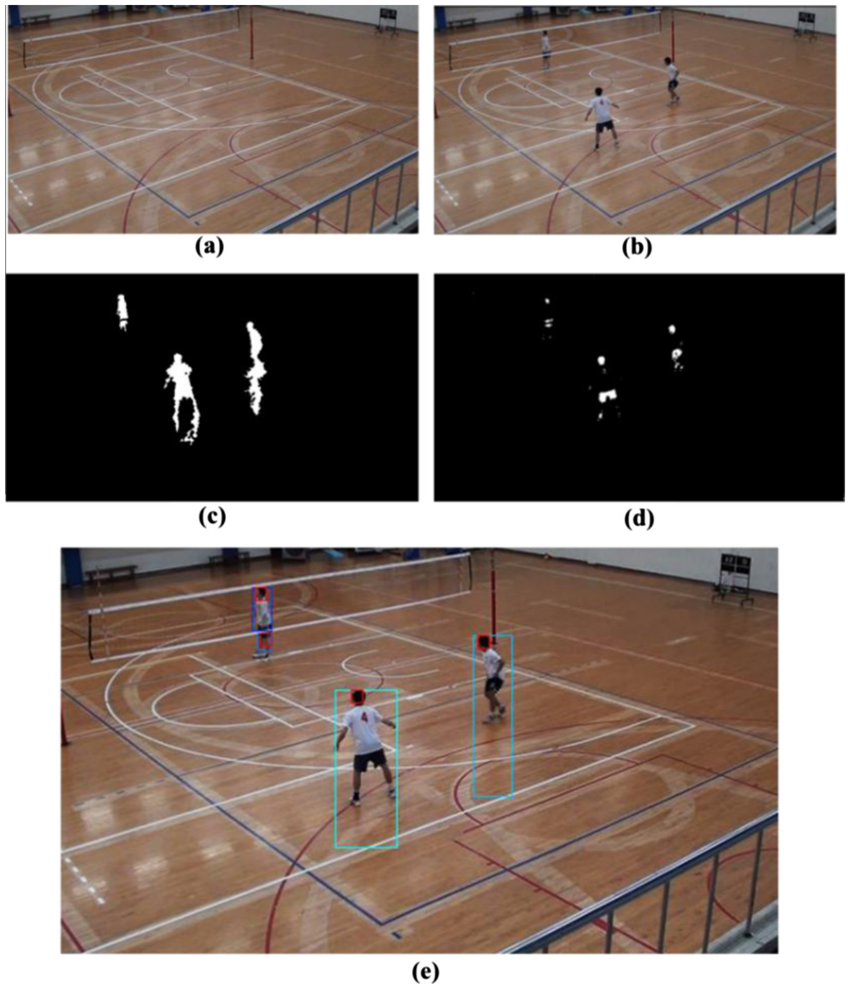

Fig. 3. Illustration of head detection. (a) Background model. (b) Original frame. (c) Segmented foreground objects. (d) Hair-colored pixels. (e) Result of head detection.

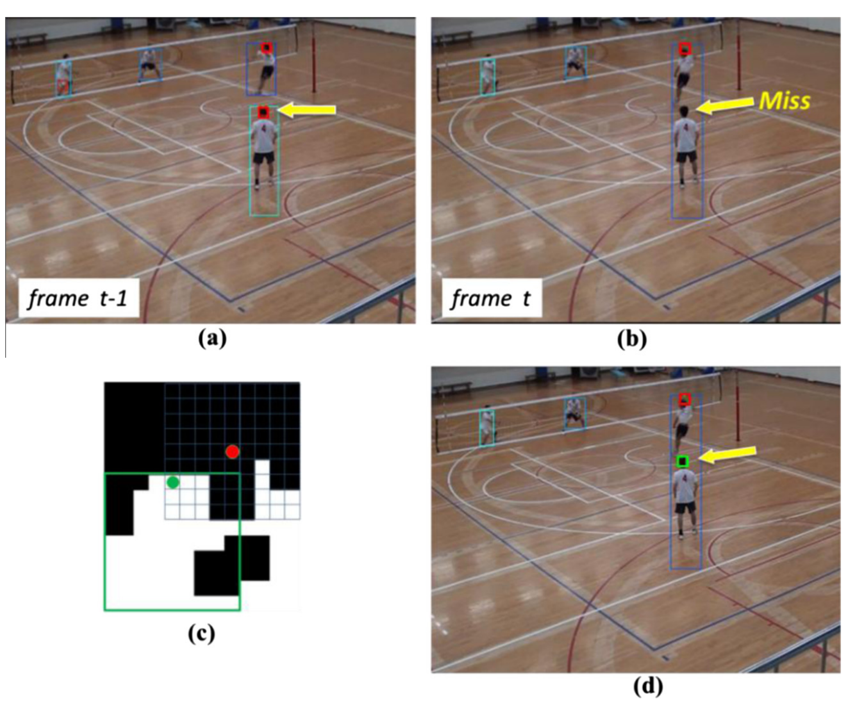

Fig. 4. Head position adjustment. (a) and (b) Consecutive video frames. (c) Search window for skin- and hair-colored pixels. (d) Adjusted head position (green rectangle). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

deo frames to the real-world coordinates or vice versa. Since the volleyball court is planar, the mapping from a position $\mathbf{p} = (x, y, 1)^T$ in the court model coordinate system to the image coordinates $\mathbf{p}' = (u, v, 1)^T$ can be described by a plane-to-plane mapping (a homography) $\mathbf{p}' = \mathbf{Hp}$, where $\mathbf{H}$ is a 3×3 homography transformation matrix [26], i.e.,

$$\begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = \begin{pmatrix} h_{00} & h_{01} & h_{02} \\ h_{10} & h_{11} & h_{12} \\ h_{20} & h_{21} & h_{22} \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \tag{1}$$

Since homogeneous coordinates are scale-invariant, we can reduce the degrees of freedom for the matrix $\mathbf{H}$ to only eight. To compute the eight independent parameters, we need at least four pairs of *corresponding points*—the points whose real world coordinates and image coordinates are both known.

As illustrated in Fig. 5, we set four reference poles, which are 100 cm in height and perpendicular to the ground, in the court for camera calibration. Taking the top and foot points of the reference poles as corresponding points, we compute the homography transformation $\mathbf{H}_0$ between the image plane and the real-world ground plane as well as the $\mathbf{H}_{100}$ between the image plane and the real-world plane at a height of 100 cm. Assuming that player height information is available, now we can derive the homography transformation $\mathbf{H}_\mathbf{p}$ between the image plane and the real-world plane at the height of the player as a linear combination of $\mathbf{H}_0$ and $\mathbf{H}_{100}$. With $\mathbf{H}_\mathbf{p}$, the head positions tracked over consecutive frames can be mapped onto the court plane to produce the moving trajectory of the player.

3. Jump pattern recognition

In this section, we attempt to recognize jump patterns in the player trajectory derived in the previous section and differentiate jump tracks from other planar moving tracks. A physics-based validation approach is also proposed to identify reasonable jump tracks and filter out abnormal ones.

3.1. Observation

Consider the player location along a trajectory under the constant height assumption. As a player is jumping above the ground plane, the player seems to have a sudden shift in location in the "planar" trajectory, as shown in Fig. 6. This phenomenon is likely to cause troubles in trajectory-based event detection if the associated jump tracks are not differentiated from other planar moving tracks.

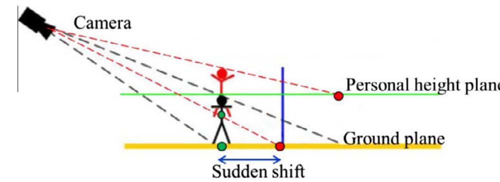

Fig. 6. Illustration of the sudden shift in location when the player jumps.

To investigate the characteristics of jump actions, a video clip of a person walking counterclockwise in a room with jumps at five locations is considered. Fig. 7(a) shows the initial frame of the video. The reconstructed top-view image of the moving trajectory is shown in Fig. 7(b), where red points indicate jump locations. It can be seen that jumps cause pulse-like tracks in the trajectory. An important characteristic of such tracks is that their extensions intersect at the *vanishing point* of vertical lines in the scenes, which is also the vertically projected camera location, as shown in Fig. 7(c).

3.2. Jump track detection

Based on the foregoing observation, we adopt the distance from the above mentioned vanishing point to each tracked player position as the main feature for jump track detection. Fig. 8 illustrates the schematic diagram of jump track detection.

In the proposed approach, for each calibrated player location $L(t)$, the distance $d(t)$ to the vanishing point C is computed, as shown in Fig. 8(a) and (b), where t represents the elapsed time. For noise reduction, Gaussian smoothing is applied to $d(t)$. Since a jump typically causes a sudden increase followed by a sudden decrease in the distance from C to $L(t)$, a peak in $d(t)$ corresponds to a possible jump. Thus, we search for zero-crossing points of $d'(t)$—the first derivative of $d(t)$, and locate *peaks* at the zero-crossing points of "positive-to-negative." Based on kinematics, a player would squat slightly before and after a jump. Thus, to determine the start and end points of a jump track, we locate the nearest *valleys* before and after each peak (termed V_b and V_a, respectively) by finding the zero-crossing points of "negative-to-positive," as shown in Fig. 8(c).

Assume P, V_b, and V_a are associated with time t_p, t_b, and t_a, respectively. According to physics, rising and falling in a jump should take the same time, i.e., $t_p - t_b = t_a - t_p$. However, in some cases, the player may move away from the vanishing point before a jump, which causes continuously increase in $d(t)$ toward the peak of a jump, as exemplified in Fig. 9(a) that the precedent valley

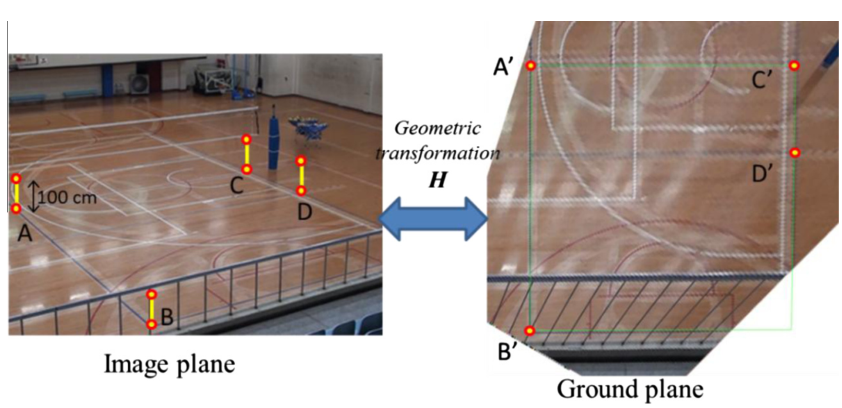

Fig. 5. Camera calibration using four reference poles.

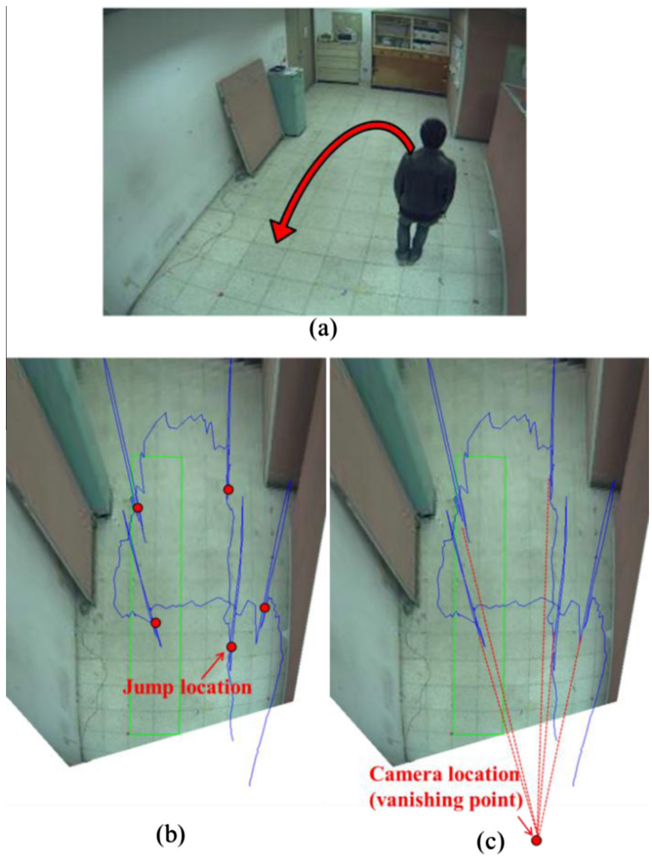

Fig. 7. Illustration of jump patterns. (a) Initial frame of a video clip of a person walking counterclockwise in a room with jumps at five locations. (b) Moving trajectory mapped on the ground plane. (c) Extensions of jump tracks and the vanishing point.

point is located away from the jump track. Fig. 9(b) shows the player trajectory with the red part indicating the detected jump track. One can see that the track before the jump where the player moves away from the vanishing point is mistakenly identified as a part of the jump track due to the incorrect location of the valley point V_b.

To resolve the foregoing problem, we refine our approach of finding the start and end points of a jump track, as illustrated in Fig. 10(a). For a peak P, we find the nearest valley V_1, e.g., at time $t_p + \Delta t$, and then we locate another feature point V_2 on the opposite side of P, e.g., at time $t_p - \Delta t$. The pattern "$V_1 - P - V_2$" can be regarded as a *jump candidate* if the temporal distance Δt between P and V_1 (or V_2) is within a range $[\tau_1, \tau_2]$. The determination of $[\tau_1, \tau_2]$ along with the physics-based jump pattern validation will be presented in Section 3.4. Fig. 10(b) shows the rectified result of jump track extraction, where we can see that the track before the jump as the player moves away from the vanishing point is no longer considered as a part of the jump track.

3.3. Jump point locating and jump height estimation

With the start and end points (V_1 and V_2) of a jump track, we can determine the "jump point"—the player location mapped onto the ground plane when the player reaches the top during a jump. Based on the symmetry of V_1 and V_2 (as mentioned in the previous section), we connect V_1 and V_2 on the player trajectory and consider the midpoint to be the jump point, as shown in Fig. 11.

On the other hand, the information of jump height is of vital importance in semantic/tactic analysis of volleyball video since the higher a player jumps, the better chance of scoring he/she has. Fig. 12 presents the illustration of jump height estimation based on the geometry, where A is the position of the camera projected onto the personal height plane, B is the jump point, C is the camera location, and D is the peak point of the jump track. Since

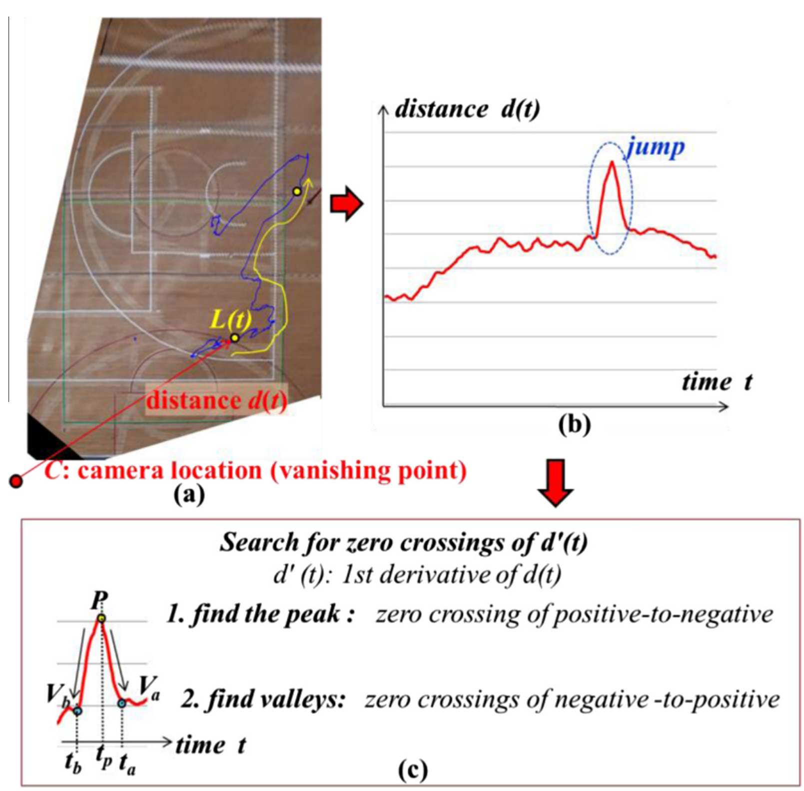

Fig. 8. Schematic diagram of jump track detection.

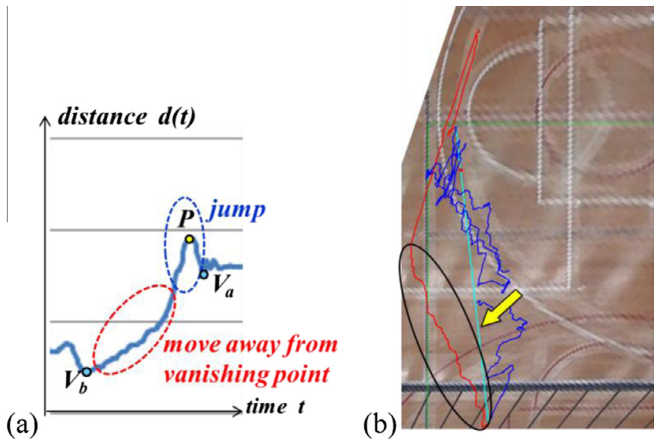

Fig. 9. Example of a player moving away from the vanishing point before a jump. (a) Distance function $d(t)$ and feature points P, V_b, and V_a. (b) Player trajectory, where the red part indicates the extracted jump track. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

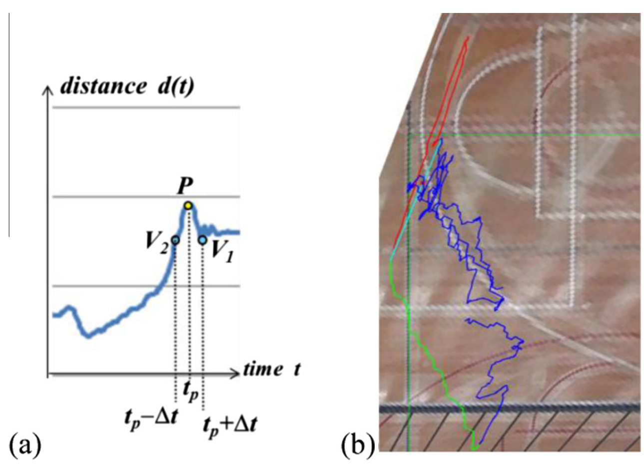

Fig. 10. Illustration for refined approach of finding the start and end points of a jump track. (a) Distance function $d(t)$ and feature points P, V_1, and V_2. (b) Player trajectory, where the red part indicates the extracted jump track. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

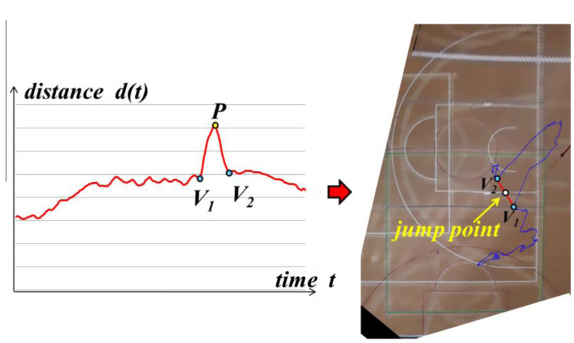

Fig. 11. Example of jump point locating.

the 3D camera position and the player height are known, we can calculate the height difference h_C between the camera and a player. Then, the jump height h_J can be estimated by

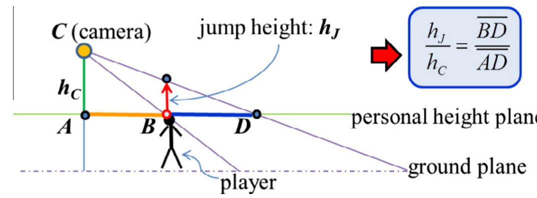

Fig. 12. Illustration of jump height estimation.

$$h_J = h_c \times \frac{\overline{BD}}{\overline{AD}} \tag{2}$$

3.4. Physics-based jump pattern validation

The foregoing processing steps detect jump candidates via identification of pulse-like patterns in the player trajectory. However, similar patterns which are not associated with jumps may also be extracted. Hence, we propose a physics-based validation approach to filter out false candidates of jump tracks.

In theory, the relation between the jump height h and the time duration of the player jumping in the air is based on the theorem of gravitational acceleration and Newton's Second Law of Motion:

$$h = \frac{1}{2} g \times \Delta t^2 \tag{3}$$

where g is the acceleration of gravity, and Δt is the time duration from the peak P to the nearest valley, V_1 or V_2, (see Fig. 10). Initially, we set the minimum jump height h_{min} = 20 cm, that is, it cannot be regarded as a jump if the player leaves the ground at a distance of no more than 20 cm. As far as is known, the greatest jump height of a professional volleyball player is about 120 cm. So, we set the maximum jump height h_{max} = 130 cm. By Eq. (3) we can derive the range $[\tau_1, \tau_2] = [0.202, 0.515]$ (in second), or approximately [6,16] (in frame difference) to judge whether a "$V_1 - P - V_2$" pattern can be a jump candidate. To meet the practical requirement and specific user needs, our proposed system allows user to designate the range of jump height, and accordingly the range $[\tau_1, \tau_2]$.

For jump pattern validation, we compute the deviation E between the jump height h_J estimated in the previous section and the jump height h derived by the theorem of gravitational acceleration, i.e.,

$$E = h_J - h = h_J - \frac{1}{2} g \times \Delta t^2 \tag{4}$$

A jump candidate is deemed *false* if its deviation E is larger than a threshold. With the aid of the physical characteristics, several false jump candidates can be filtered out, which greatly enhances the effectiveness of the proposed jump pattern recognition scheme.

4. Experimental results and discussion

To evaluate the effectiveness of the proposed head tracking and jump pattern recognition approaches, we conduct experiments on the video data (MPEG-2, 640 × 488, 29.97 fps) of 2-on-2 volleyball games played by the members of the school volleyball team of National Chiao Tung University, Taiwan. Ten offensive clips (3559 frames in total) are manually selected to test the proposed methods. Our system is implemented in C++ with OpenCV libraries [27], and run on a Dell notebook (Intel Core i5-3317U CPU @1.70 GHz, 8 GB RAM, Windows 7 64-bit OS).

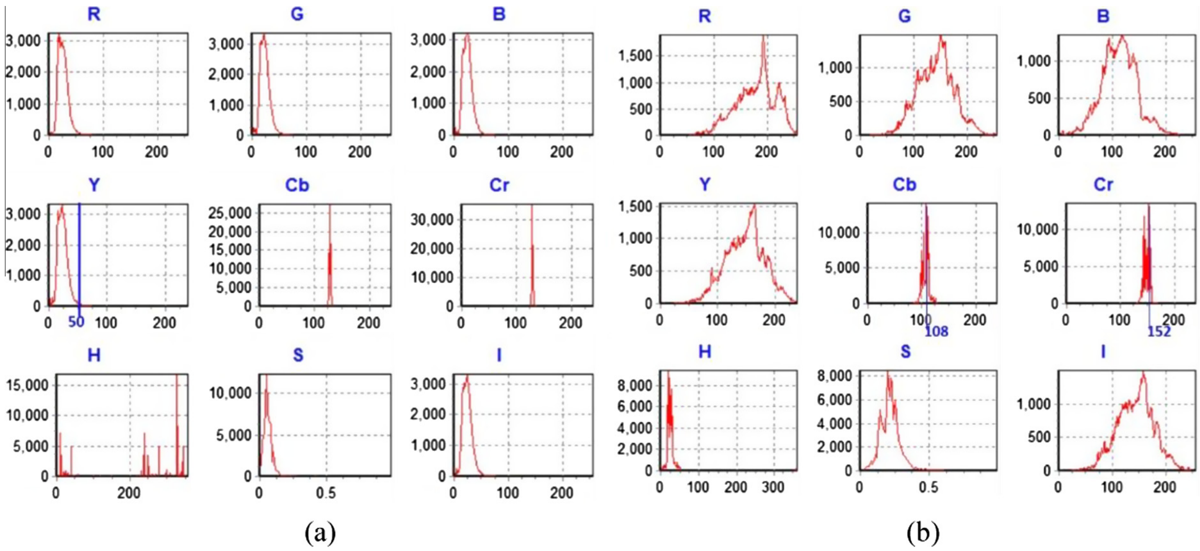

Fig. 13. Color histograms of (a) head images and (b) skin images.

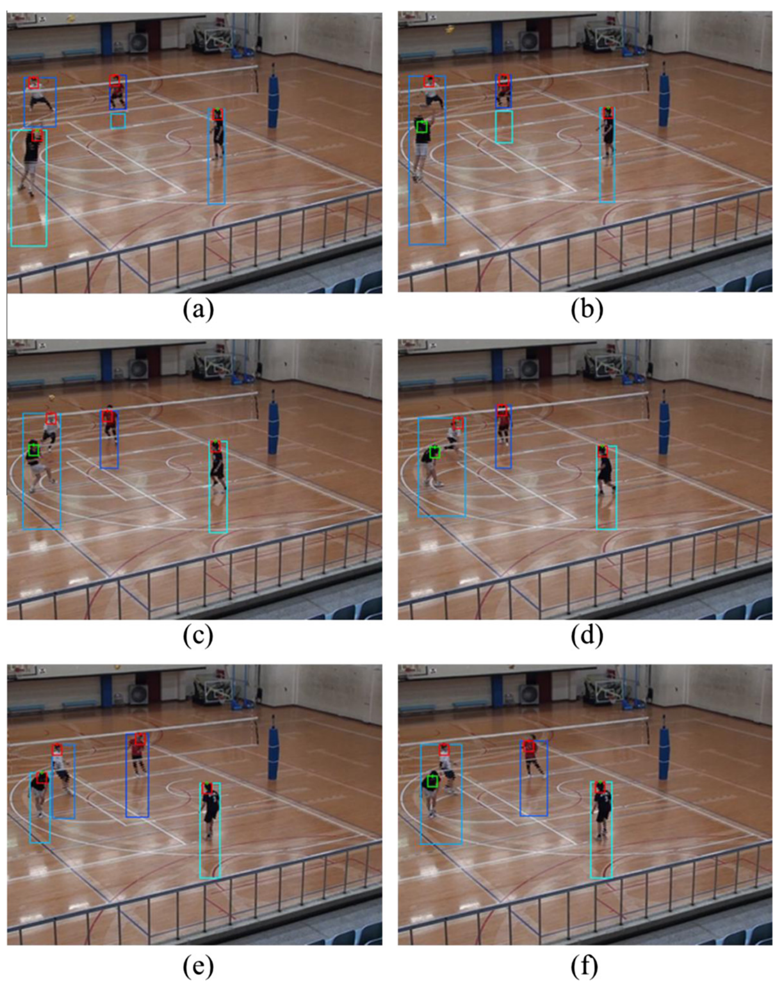

Fig. 14. Result of head position detection with adjustment for frames #343, #349, #373, #384, #396, and #412 of a test clip, where red rectangle indicates the initially detected head positions and green rectangles indicate the head positions after adjustment. (a) and (e) The head positions are detected correctly without adjustment. (b)–(d) and (f) The two players on the left interfere with each other but reasonable head positions can be obtained through the process of head position adjustment. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

4.1. Parameter setting

For player trajectory computation, the hair and skin colors are used as features. The ranges of hair and skin colors are determined statistically. To obtain the color distribution of the hairs (skins) in video frames, 50 hair (skin) images are segmented manually from various videos to produce the color histograms in RGB, YCbCr, and HSI color spaces, as shown in Fig. 13. Due to its superior ability to discriminate the above color features, YCbCr color space is chosen. Assuming that the hair color is black, the hair color range is set to $Y < 50$, which covers 99% of the pixels of the 50 hair images, as shown in Fig. 13(a). For the skin color, peaks of Cb and Cr histograms in Fig. 13(b) are at 108 and 152, respectively. Centered at the peaks, the Cb and Cr ranges of the skin color are determined to cover 99% of the pixels of the 50 skin images, i.e., $90 \leqslant Cb \leqslant 126$ and $133 \leqslant Cr \leqslant 171$ in our experiments.

4.2. Results of head tracking and player trajectory computation

Since head positions may not be detected accurately in all frames, head position adjustment is required. Fig. 14 demonstrates the effect of head position adjustment, where red rectangles indicate the detected head positions and green rectangles indicate the head positions after adjustment. In Fig. 14(a) and (e), all the head positions are detected correctly. In Fig. 14(b)–(d) and (f), occlusions occur between the two players on the left, resulting in incorrect segmentation of foreground objects. Through the process of head position adjustment, reasonable head positions, as indicated by the green rectangles, can be obtained. In general, the color of the player's jersey may influence the result of head position adjustment. For example, in Fig. 14(b), (c) and (f), the black jersey, being close to the hair region in color, results in some deviation in the head positions after adjustment.

For performance evaluation, an experienced volleyball player is asked to manually check the head tracking results (on the basis of *hit* and *miss*) of each frame in the test video clips. A hit or a miss means the head position of a player is detected correctly or incorrectly, respectively, in a frame, and the accuracy is defined by Eq. (5).

$$Accuracy = \frac{\#hit}{\#hit + \#miss} \tag{5}$$

Statistics of the accuracy of the proposed head tracking and adjustment algorithm for two players are shown in Fig. 15 for a total of ten test clips, where the horizontal axis indicates the clip ID, the dotted (solid) line shows the accuracy results obtained without

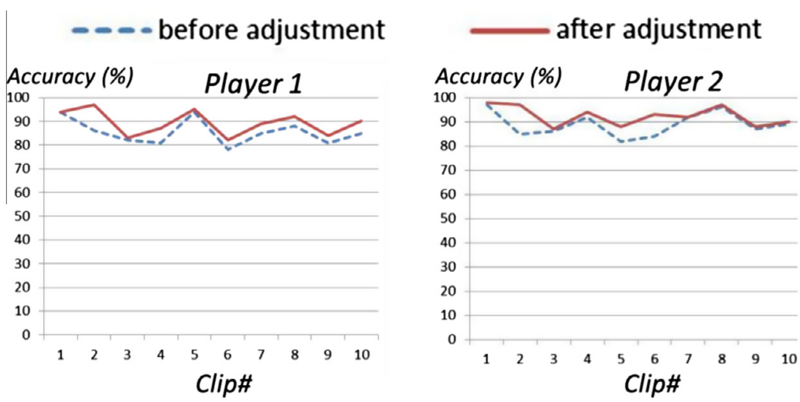

Fig. 15. Accuracy of the proposed head tracking and adjustment algorithm for test clips.

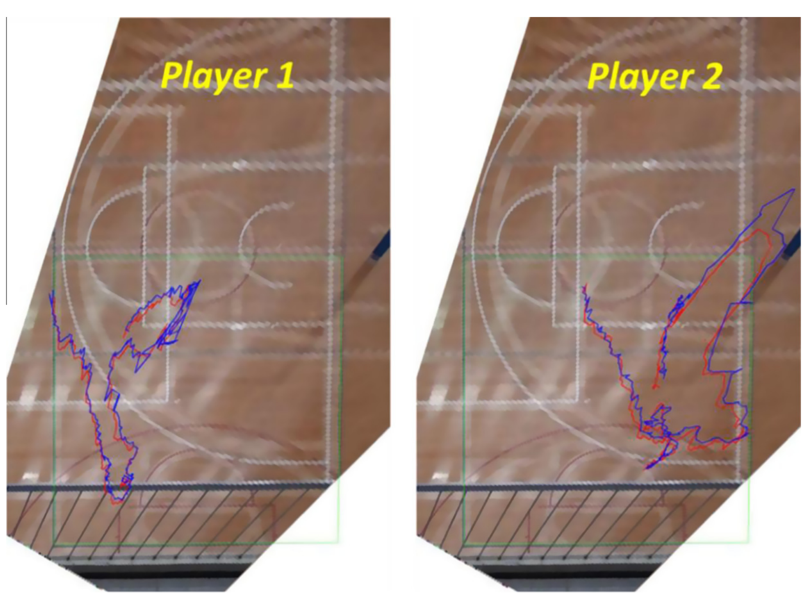

Fig. 16. Visual comparison between the player trajectory produced by our method (in blue) and the manually labeled trajectory (in red) for test clip #1. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

Table 1
Performance of jump pattern recognition.

	Player 1	Player 2	**Overall**
#total	10	11	**21**
#correct	9	10	**19**
#false	2	1	**3**
Precision (%)	81.82	90.91	**86.36**
Recall (%)	90.00	90.91	**90.48**

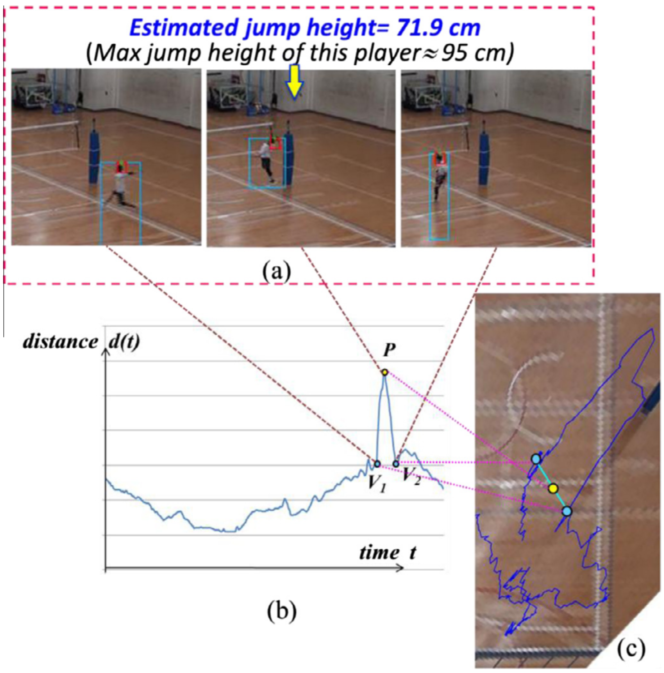

Fig. 17. An example of jump pattern recognition.

(with) the process of head adjustment. Overall, the average accuracy rate (after adjustment) is about 90%. Fig. 16 presents the visual comparison between the player trajectories produced by our method (in blue) and the manually labeled trajectories (in red) for two players in test clip #1. Although the two trajectories do not coincide completely, the main directions of motion of the trajectories are matched very well.

4.3. Results and discussion of jump pattern recognition

The performance of jump pattern recognition is presented in Table 1, where the row "#*total*" indicates the total number of jumps in the video clips. The terms "#*correct*" and "#*false*" represent the numbers of correct recognitions and false alarms, respectively. The precision and recall are defined by Eqs. (6) and (7).

$$Precision = \frac{\#correct}{\#correct + \#false} \tag{6}$$

$$Recall = \frac{\#correct}{\#total} \tag{7}$$

The experiments show that our preliminary work of jump pattern recognition achieves encouraging results on 2-on-2 volleyball videos. The average precision and recall rates are up to 86.36% and 90.48%, respectively.

An example of jump pattern recognition is illustrated in Fig. 17. Fig. 17(a) presents the frames of V_1, P and V_2, which are also marked along the distance curve shown in Fig. 17(b). It is easy to see that V_1 and V_2 are the start and end points of a detected jump track and P corresponds to the peak of the jump. Fig. 17(c) shows the coordinates of V_1, P and V_2 mapped on the real-world court plane. In this example, V_1 and V_2 correctly indicate the moments at which the player is leaving the ground and then landing, respectively, and P indicates the moment when the player is reaching the highest point to hit the ball. More results of jump pattern recognition are demonstrated in Figs. 18–21.

Fig. 18(a) shows the player trajectory projected on the court plane. The planar moving track of the player trajectory is produced by cutting the jump track (from V_1 to V_2) and connecting V_1 and V_2 with a line segment, as presented in Fig. 18(b). Fig. 18(c) shows the close-ups of the jumping player at the frames of V_1, P and V_2, as well as the estimated jump height. Since we are unable to measure the jump height of the player during the game, the max jump height of the player is also provided as a reference. The max jump height of the player is obtained via measuring the distance between the highest point the player can reach when he stands on the ground and the point when he jump to the best of his ability. In general, the jump height of the player during the game will be smaller than his max jump height because instead of an upward jump of maximum height, he should consider the ball motion and jump with a particular height/direction when attempting to hit the ball. That is the reason why the corresponding shift in location does not always result in a sharp peak in the player trajectory.

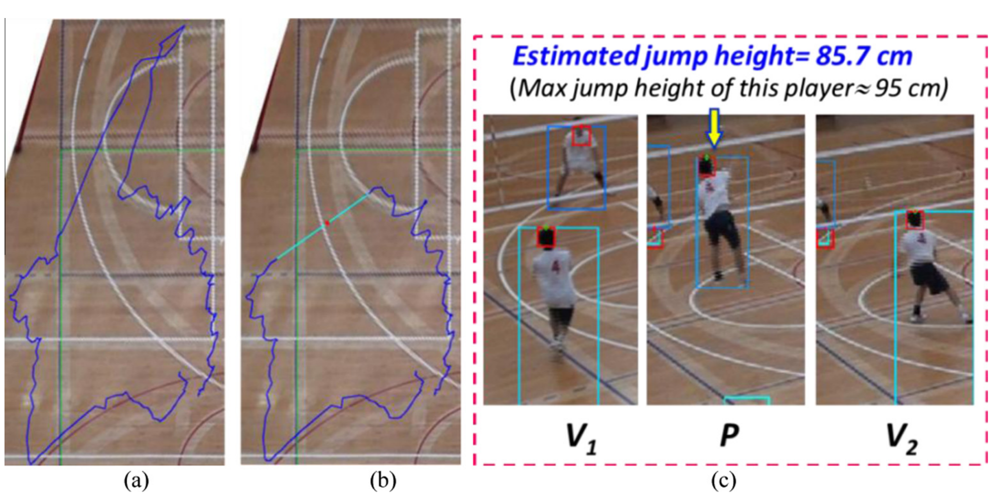

Fig. 18. Demonstration of jump pattern recognition. (a) Player trajectory projected on the court plane. (b) Planar moving trajectory without the jump track. (c) Close-ups of the jumping player at the frames of V_1, P, and V_2, as well as the estimated jump height.

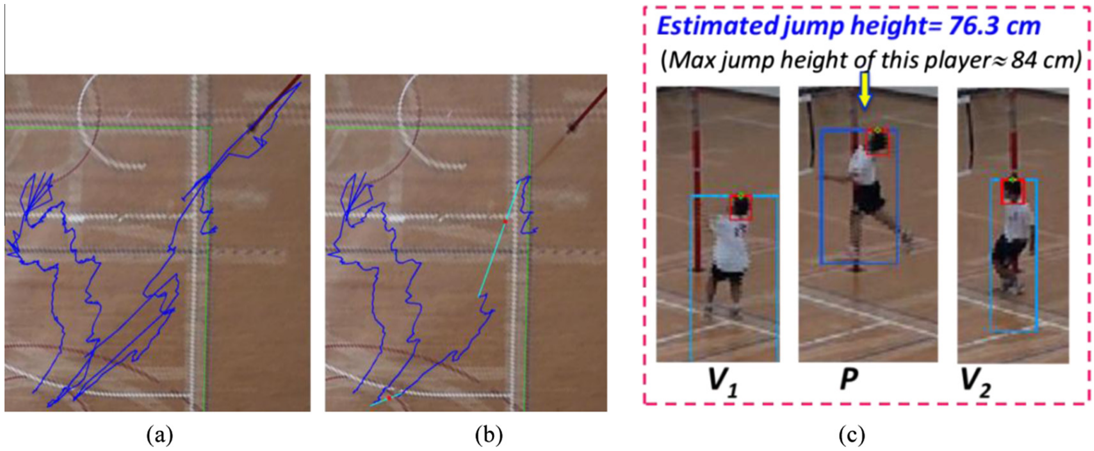

Fig. 19. Demonstration of jump pattern recognition. (a) Player trajectory projected on the court plane. (b) Planar moving trajectory without the jump track. (c) Close-ups of the jumping player at the frames of V_1, P, and V_2, as well as the estimated jump height.

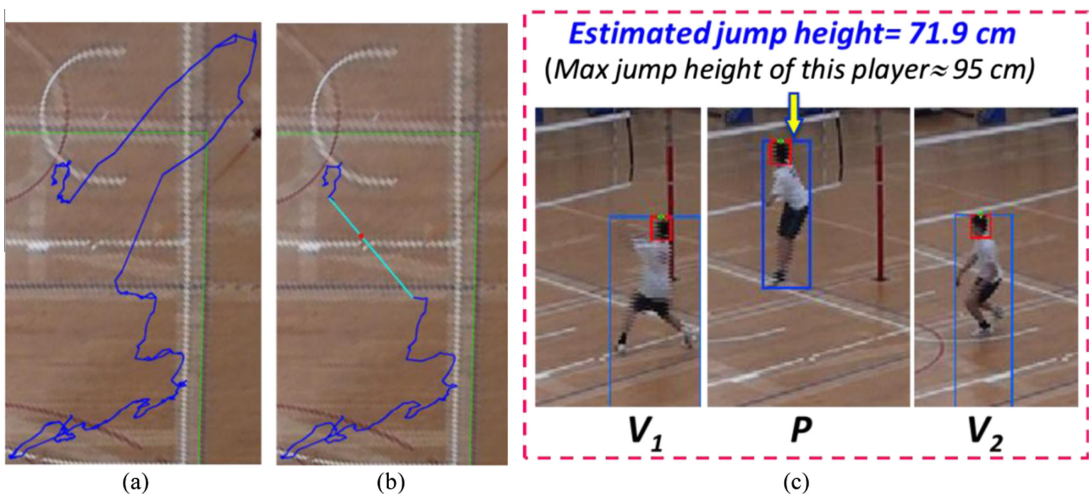

Fig. 20. Demonstration of jump pattern recognition. (a) Player trajectory projected on the court plane. (b) Planar moving trajectory without the jump track. (c) Close-ups of the jumping player at the frames of V_1, P and V_2, as well as the estimated jump height.

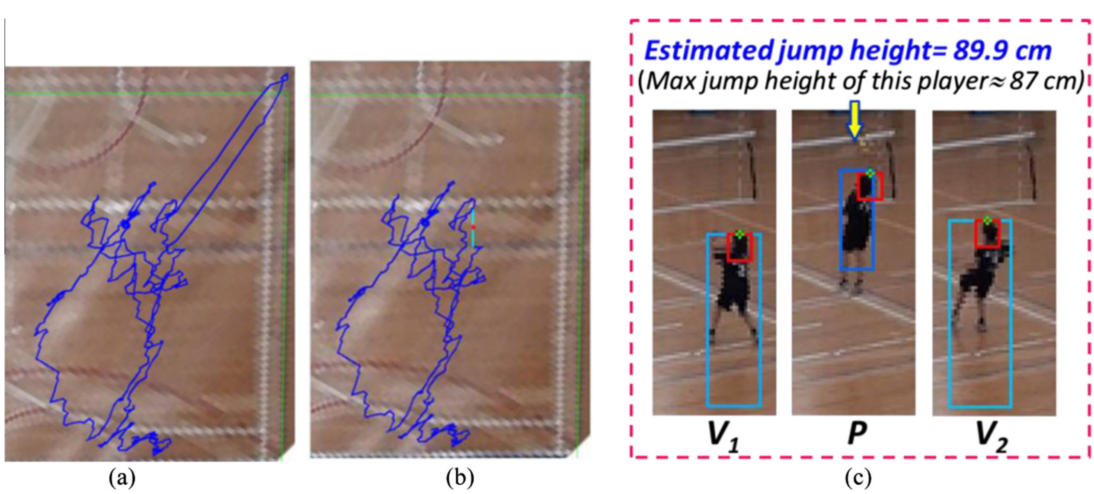

Fig. 21. Demonstration of jump pattern recognition. (a) Player trajectory projected on the court plane. (b) Planar moving trajectory without the jump track. (c) Close-ups of the jumping player at the frames of V_1, P, and V_2, as well as the estimated jump height.

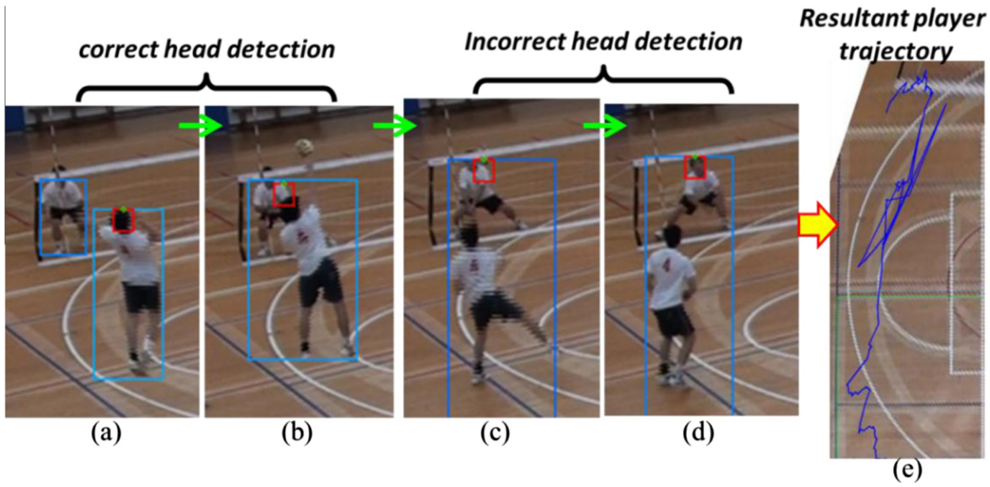

Fig. 22. Illustration of an error case of jump pattern recognition. (a) The head of the jumping player is correctly detected. (b) The jumping player is merged with the opponent player. (c) and (d) Head detection is incorrect. (e) The resultant player trajectory.

Although it may not be so accurate, the results of jump height estimation in Figs. 18–20 are reasonable and acceptable. As for Fig. 21, the estimated jump height of the player is more than his max jump height, which is not correct.

Note that inaccurate player trajectory is the major cause of incorrect results of jump pattern recognition and jump height estimation, which may due to errors in camera calibration and image feature extraction, etc. An error case is shown in Fig. 22. Initially, the head of the jumping player is correctly detected, as shown in Fig. 22(a). Later on, the player is merged with the opponent player in Fig. 22(b) resulting in erroneous head detection and player trajectory, as shown in Fig. 22(c), (d) and (e), respectively, causing the failure in jump pattern recognition. In this paper, we choose a less complex case of the 2-on-2 volleyball video, wherein player occlusion rarely occurs, to test and verify our proposed jump pattern recognition approach. For more complex cases with frequent player occlusion, such as the 6-on-6 volleyball video and 5-on-5 basketball video, more robust player tracking, trajectory computation, and even shadow removal approaches may be required. We will focus on such cases in our future work.

5. Conclusion

For video analysis and content understanding, human moving trajectory is one of the most informative representations. However, a jump action typically leads to a violent change in human moving trajectory, since the person suddenly leaves the original plane on which he/she has been moving. The abnormal changes in the trajectory might cause errors in trajectory-based video analysis. Hence, differentiating jump tracks from planar moving tracks is indispensable. Moreover, jump actions typically imply significant events, especially in sports games.

In this paper, we design a jump pattern recognition approach by utilizing the important characteristic that extensions of pulse-like tracks caused by jumps in the trajectory will go through a vanishing point. Furthermore, we propose a physics-based validation, which considers the relation between the jump height and the time duration while the player is in the air, based on the theorem of gravitational acceleration and Newton's Second Law of Motion. Many false jump candidates can be filtered out through this validation process. The proposed framework is experimented on 2-on-2 volleyball video clips and encouraging results of 86.36% and 90.48% are obtained for the precision and recall rates, respectively, for jump pattern recognition. It is our belief that the recognized jump pattern, the located jump position and the estimated jump height will greatly assist the existing or even the oncoming research on trajectory-based video analysis.

Acknowledgments

This research is supported in part by projects ICTL-102-Q707, ATU-102-W958, NSC-98-2221-E-009-124-MY2, and NSC-102-2221-E-009–031. Thanks to the members of the school volleyball team of National Chiao Tung University, Taiwan, especially Mr. Chen-Hsing Chang, who is a physical education teacher and is also a professional volleyball coach.

References

[1] S.-F. Chang, The holy grail of content-based media analysis, IEEE Multimedia 9 (2) (2002) 6–10.
[2] A.G. Money, H. Agius, Video summarisation: a conceptual framework and survey of the state of the art, Journal of Visual Communication and Image Representation 19 (2) (2008) 121–143.
[3] R.O. Preda, N.D. Vizireanu, Robust wavelet-based video watermarking scheme for copyright protection using the human visual system, Journal of Electronic Imaging 20 (1) (2011). 013022–1–8.
[4] R.O. Preda, N.D. Vizireanu, Quantisation-based video watermarking in the wavelet domain with spatial and temporal redundancy, International Journal of Electronics 98 (3) (2011) 393–405.
[5] J.-S. Lee, F.D. Simone, T. Ebrahimi, Efficient video coding based on audio-visual focus of attention, Journal of Visual Communication and Image Representation 22 (8) (2011) 704–711.
[6] A. Kokaram, N. Rea, R. Dahyot, M. Tekalp, P. Bouthemy, P. Gros, I. Sezan, Browsing sports video: trends in sports-related indexing and retrieval work, IEEE Signal Processing Magazine 23 (2006) 47–58.
[7] L.-Y. Duan, M. Xu, T.-S. Chua, Q. Tian, C.-S. Xu, A mid-level representation framework for semantic sports video analysis, in: Proceedings of 11th ACM International Conference on Multimedia, 2003, pp. 33–44.
[8] C. Piciarelli, G.L. Foresti, L. Snidaro, Trajectory clustering and its applications for video surveillance, in: Proceedings of IEEE Conference Advanced Video and Signal Based Surveillance, 2005, pp. 40–45.
[9] C.-W. Su, H.-Y.M. Liao, H.-R. Tyan, C.-W. Lin, D.-Y. Chen, K.-C. Fan, Motion flow-based video retrieval, IEEE Transactions on Multimedia 9 (6) (2007) 1193–1201.
[10] F.I. Bashir, A.A. Khokhar, D. Schonfeld, Real-time motion trajectory-based indexing and retrieval of video sequences, IEEE Transactions on Multimedia 9 (1) (2007) 58–65.
[11] J.-W. Hsieh, S.-L. Yu, Y.-S. Chen, Motion-based video retrieval by trajectory matching, IEEE Transactions on Circuits and Systems for Video Technology 16 (3) (2009) 396–409.
[12] G. Zhu, Q. Huang, C. Xu, Y. Rui, S. Jiang, W. Gao, H. Yao, Trajectory based event tactics analysis in broadcast sports video, in: Proceedings of 15th ACM International Conference on Multimedia, 2007, pp. 58–67.
[13] X. Yu, H.W. Leong, C. Xu, Q. Tian, Trajectory-based ball detection and tracking in broadcast soccer video, IEEE Transactions on Multimedia 8 (6) (2006) 1164–1178.

[14] X. Yu, C. Xu, H.-W. Leong, Q. Tian, Q. Tang, K.-W. Wan, Trajectory-based ball detection and tracking with applications to semantic analysis of broadcast soccer video, in: Proceedings of 11th ACM International Conference on Multimedia, 2003, pp. 11–20.
[15] J. Han, D. Farin, P.H.N. de With, Broadcast court-net sports video analysis using fast 3-D camera modeling, IEEE Transactions on Circuits and Systems for Video Technology 18 (11) (2008) 1628–1638.
[16] J. Han, D. Farin, P.H.N. de With, A mixed-reality system for broadcasting sports video mobile devices, IEEE Multimedia 18 (2011) 72–84.
[17] Y. Luo, T.-D. Wu, J.-N. Hwang, Object-based analysis and interpretation of human motion in sports video sequences by dynamic Bayesian networks, Computer Vision and Image Understanding 92 (2) (2003) 196–216.
[18] G. Zhu, Q. Huang, C. Xu, L. Xing, W. Gao, H. Yao, Human behavior analysis for highlight ranking in broadcast racket sports video, IEEE Transactions on Multimedia 9 (6) (2007) 1167–1182.
[19] G. Zhu, C. Xu, Q. Huang, W. Gao, L. Xing, Player action recognition in broadcast tennis video with application to semantic analysis of sports game, in: Proceedings of 14th ACM International Conference on Multimedia, 2007, pp. 431–440.
[20] 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, Journal of Visual Communication and Image Representation 20 (3) (2009) 204–216.
[21] H.-T. Chen, W.-J. Tsai, S.-Y. Lee, J.-Y. Yu, Ball tracking and 3D trajectory approximation with applications to tactics analysis from single-camera volleyball sequences, Multimedia Tools and Applications 60 (3) (2011) 641–667.
[22] 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, Journal of Information Science and Engineering 24 (1) (2008) 143–157.
[23] M.-C. Hu, M.-H. Chang, J.-L. Wu, L. Chi, Robust camera calibration and player tracking in broadcast basketball video, IEEE Transactions on Multimedia 13 (2) (2011) 266–279.
[24] M. Thomas, C. Koch, M. Tilp, B. Horst, Visual tracking of athletes in beach volleyball using a single camera, International Journal of Computer Science in Sport 6 (2007) 21–34.
[25] Z. Zivkovic, Improved adaptive Gaussian mixture model for background subtraction, in: Proceedings of 17th International Conference on Pattern Recognition, vol. 2, 2004, pp. 28–31.
[26] D. Farin, S. Krabbe, P.H.N. de With, W. Effelsberg, Robust camera calibration for sport videos using court models, in: Proceedings of Storage and Retrieval Methods and Applications for Multimedia, vol. 5307, 2004, pp. 80–91.
[27] Open source computer vision [Online]. Available: <http://opencv.org/>.