


Player Tracking using Multi-viewpoint Images in Basketball Analysis

Shuji Tanikawa and Norio Tagawa ^a

Graduate School of Systems Design, Tokyo Metropolitan University, Hino, Tokyo 191-0065, Japan

Keywords: Basketball Analysis, Player Tracking, Occlusion Avoidance, Multiple Camera, Radon Transform, OpenPose.

Abstract: In this study, we aim to realize the automatic tracking of basketball players by avoiding occlusion of players, which is an important issue in basketball video analysis, using multi-viewpoint images. Images taken with a hand-held camera are used, to expand the scope of application to uses such as school club activities. By integrating the player tracking results from each camera image with a 2-D map viewed from above the court, using projective transformation, the occlusion caused by one camera is stably solved using the information from other cameras. In addition, using OpenPose for player detection reduces the occlusion that occurs in each camera image before all camera images are integrated. We confirm the effectiveness of our method by experiments with real image sequences.

1 INTRODUCTION

Research on sports video analysis has been conducted actively in recent years, with the aim of improving individual players' ability and team competitiveness, and providing effective information to audiences (Xu et al., 2004)(Vaeyens et al., 2007). In the field of basketball analysis, conventionally, there has been detailed study of video processing after an individual player is cut out: for example, analysis of shoot forms (Liu et al., 2011) and elucidation of the mechanics of the human body when an injury occurs (Krosshaug et al., 2007). In addition, in recent years, there has been advanced research on analysis of team play, such as screen play and pick and roll (Chen et al., 2009)(Chen et al., 2012)(Fu et al., 2011)(Liu et al., 2011)(Lucey et al., 2014)(Liu et al., 2013)-(Lucey et al., 2014). These studies often analyze a non-occlusion image taken from above the court, to easily realize and use player tracking (see Fig. 1), or the player tracking results may be processed manually by analysts. For football videos, many stable player tracking methods using probabilistic techniques, such as the Kalman filter or particle filter, have been proposed. In basketball, in contrast to soccer, occlusion between players occurs frequently because of differences in the size of the court and the camera viewpoints, so a practical method for automatic player tracking has not yet been established.

We do not assume a special environment—for ex-

ample, a stadium with multiple cameras placed on the ceiling—to enable the player tracking method to be applicable to club activities in high school and junior high school. For these purposes, it is desirable to process images that are captured by hand-held cameras from the side, or obliquely above the court (Wen et al., 2016)(Hu et al., 2011). In this case, occlusion between players is extremely likely to occur. Therefore, we consider the use of multiple videos taken from different viewpoints, which potentially allows players hidden from one camera to be captured by another camera. In addition, as another advantage of using multiple viewpoints, acquisition of three-dimensional information about players becomes possible.

In this study, we assume three cameras with different viewpoints, and propose a method to integrate the information about the players' positions obtained by them appropriately. Because each camera continually changes its viewpoint, it is necessary to perform calibration efficiently, assuming real-time processing. When detecting a player in each image, it is desirable to use a method that is effective even in the presence of occlusion. Integrating the information from each camera requires a procedure to ensure sufficient stability when the player is hidden from a particular camera or when the player is detected again. In this paper, we propose a player tracking system that satisfies the above requirements. The first feature of the proposed method is to use OpenPose, a human joint detection method based on Deep Neural Network architecture, to avoid some occlusion of players. Another feature


^a  <https://orcid.org/0000-0003-0212-9265>



Figure 1: Examples of images captured above the court.

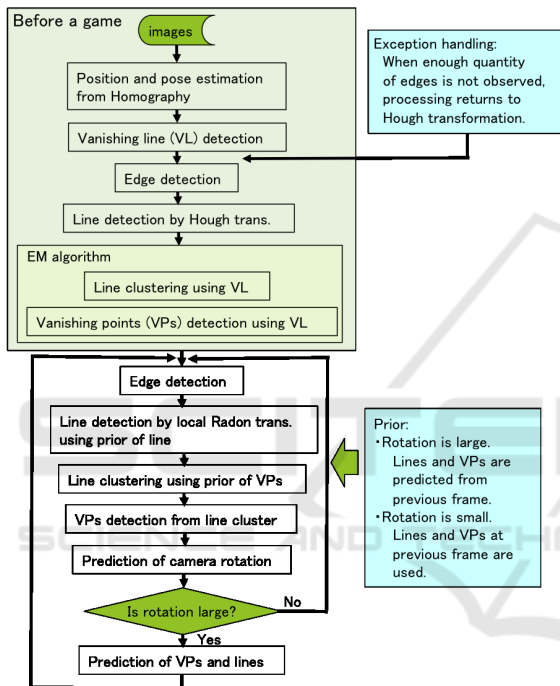


Figure 2: Outline of our camera pose estimation.

is that the position of the player detected from each camera’s viewpoint is converted into a reference image viewed from above the court, and the detection results from multiple viewpoints in the reference image are appropriately integrated. In consideration of cases where a player cannot be seen due to occlusion in a certain camera, a data structure and an algorithm capable of handling the disappearance and occurrence of a player position are developed. The performance and effectiveness of our method has been confirmed through real image experiments.

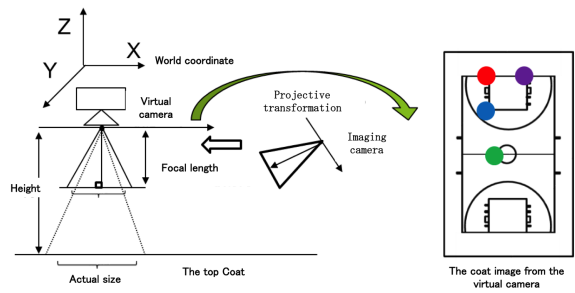


Figure 3: Camera layout and coordinates, and relation between captured image and reference image.

2 CALIBRATION OF CAMERA POSE

Our method for calibrating the camera position and direction has been proposed in (Idaka et al., 2017) in advance. This section introduces the outline shown in Fig. 2. Because a basketball game is played alternately in each half court, the camera direction moves back and forth between both half courts, depending on the offense and defense. Therefore, before starting the game, the standard position and direction corresponding to each half court should be determined. Because an image without a player can be used, feature point correspondence, using a court corner or similar feature, can easily be adopted. The homography matrix \vec{H} representing the projective transformation is determined, and decomposed into camera rotation and translation using the following equation (Kanatani, 1993).

$$k\vec{H}^T = \left[r\vec{I} + \begin{bmatrix} p \\ q \\ r \end{bmatrix} \begin{bmatrix} A & B & C \end{bmatrix} \right] \vec{R}, \quad (1)$$

where (p, q, r) denotes the plane $Z = pX + qY + r$ corresponding to the court, (A, B, C) indicates the camera position, \vec{R} indicates the camera direction, \vec{I} indicates a 3×3 unit matrix, and k is an arbitrary value. The coordinates of the virtual camera viewing a court from directly above are used as the world coordinates, and the two-dimensional map (2-D map) used in the following is defined by the image viewed by the virtual camera. Figure 3 shows the relation between the virtual camera and the actual imaging camera. The pose of the imaging camera is measured with respect to the world coordinates. The colored points in the right panel of Fig. 3 are the feature points used to determine \vec{H} in Eq. 1.

In the processing during a game, under the assumption that the camera position slightly changes,

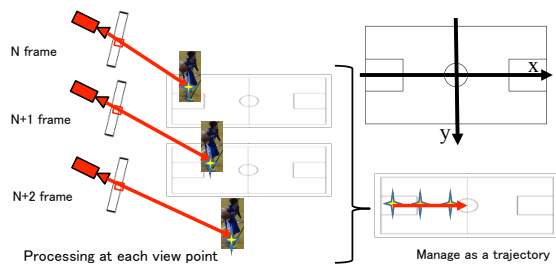


Figure 4: Generation of player's trajectory from player detection result in video from each viewpoint.

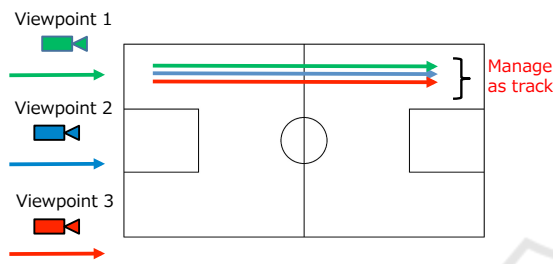


Figure 5: Trajectories from each viewpoint that are close on the 2-D map in multiple successive frames are grouped as a track.

the camera direction is estimated at each frame by detecting two orthogonal vanishing points (VPs). When the viewing direction is fixed, the camera may move because of hand movement, which is expected to be random and small. In contrast, when the viewing direction is intentionally changed, a comparatively large rotation should be considered. For a small rotation, the VPs can be detected by the Radon transform, instead of the Hough transform, to reduce the computational cost. By the Radon transform, the Hough parameter space can be evaluated locally around the parameters obtained in the previous frame. The details is explained in (Idaka et al., 2017).

3 PLAYER TRACKING METHOD

3.1 Outline of Tracking

Our tracking method has the following features:

1. The results of detecting each player from each viewpoint are projected onto a common 2-D map, and a "trajectory" corresponding to each player is determined while evaluating the temporal continuity between frames (see Fig. 4). Here, the trajectory is defined as the tracking result of the player from a single viewpoint. The tracking results from all viewpoints can be evaluated in the same coordinate system.

Trajectory length (frame number)	X	Y	Label number	Trajectory replacement log	Start frame of Trajectory	End frame of Trajectory	Correspondence log
0	68,000	3,469	-8,426	181,000	0,000	181,000	0,000
1	29,000	18,627	-4,657	182,000	0,000	182,000	0,000
2	54,000	6,187	1,248	183,000	0,000	183,000	0,000
3	68,000	7,581	1,814	184,000	0,000	184,000	0,000
4	54,000	6,898	-4,584	185,000	0,000	185,000	0,000

x Number of Frames x Three viewpoints

Figure 6: Data structure representing information of players' trajectories. Information of each player's trajectory, the start frame and end frame of the trajectory (0 indicates that tracking is in progress), and other information are organized and stored for each frame and each viewpoint. For "replacement log" and "correspondence log," refer to explanation in the text.

2. By matching the trajectories from multiple viewpoints, one consistent "track" is created for each player. Occlusion can be overcome and the trajectories of the bench players can be deleted (see Fig. 5).

The following subsections explain the details.

3.2 Player Detection

We first detect the players based on the color information of the uniform. Color information is essential to identify a team, but if multiple players on the same team approach and cause occlusion, they cannot be distinguished.

In addition, OpenPose (Cao et al., 2017) has recently been applied to various studies. OpenPose can detect human joint information from an image; because skeletal information is used, it is possible to detect the presence of a person even if some joints are hidden. In this study, we investigate to what extent the OpenPose method can avoid occlusion, compared with using only color information.

Because the obtained image includes spectators and reserve players around the court, the area of the court (plus a margin of 1 m around the court) is processed, and pixels outside the area are excluded by the projective transformation.

3.3 Player Trajectory Generation

To integrate the player detection results from each viewpoint and each frame, the detection results are projected onto a 2-D map using the camera position and direction obtained in advance. The correct detection results for a specific player in successive frames are in similar locations, both in the image and on the 2-D map. Regardless of the viewpoint, to normalize the measurement of proximity, it should be evaluated not on the image but on the 2-D map. Therefore, if



Figure 7: Data structure representing information of players' tracks. It stores the trajectory number at each viewpoint associated with each player. As the frames advance, the structure extends downward.

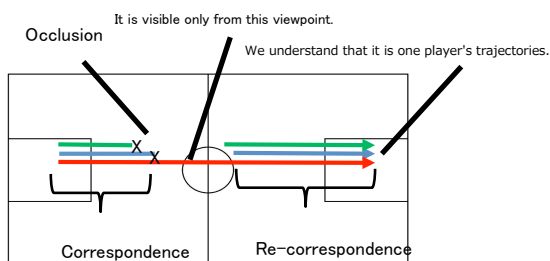


Figure 8: Correspondence of trajectories when occlusion occurs.

there are player candidates projected within 1 m, for both X and Y coordinates, on the 2-D map in successive frames, the set of points on the 2-D map is managed as a trajectory. This procedure is illustrated in Fig. 4. The threshold of 1 m is a value determined in this experiment; it is necessary to consider a systematic method to determine this threshold in the future.

Trajectories are collectively managed as an array data structure for each frame, from each viewpoint, as shown in Fig. 6. It contains information such as length, coordinates, label number (which is the management number of the trajectory), and the correspondence log (which is the number of the viewpoint to which the trajectory corresponds). Figure 6 shows the data for one team, comprising five players. If there are multiple projected player candidates within the threshold, a trajectory is created based on the coordinates of the player closest to the player in the previous frame, and "1" is placed in the replacement log after this frame. When processing the subsequent trajectory generation, if the trajectory is not associated with a trajectory obtained from another viewpoint, the trajectory is regenerated using a more distant candidate player by tracing back to the last frame having "1" in the replacement log.

3.4 Player Track Generation

The correctness of the trajectory detected from each viewpoint is confirmed by comparison with trajectories from other viewpoints. A specific player's trajectories should make the same movement on the 2-D map. If the player's trajectory projected in multiple

consecutive frames from one viewpoint exists close to the player's trajectory from another viewpoint, these trajectories are considered to correspond, and they are recognized as a collection of the player's correct trajectories. An example of a trajectory that is detected incorrectly is the trajectory of a reserve player. Even if the background is removed using the projective transform, the torso and head of the reserve player on the near side tend to remain in the image more than the reserve player on the far side of the court. These areas are difficult to cut even if we use the colors or the human joints of the players, and they are detected as trajectories as long as they continue to appear in the image, so the correspondence with other viewpoints is used to distinguish them from the correct players.

Correspondence processing is performed for all frames in temporal order. The trajectories from each viewpoint in the frame being processed are compared in a round-robin manner, and correspondence is made using coordinates in a number of successive frames; this number was defined as 10 frames in this study. If a trajectory that has been interrupted by occlusion is subsequently detected again, multiple tracks may be present nearby. If we try to map the trajectory to the track immediately, there are multiple candidates and the ambiguity is high. Therefore, the frame is advanced until the player's position on the track deviates from the other players' positions to a certain extent, and then correspondence with the track is made. In this study, we set the separation threshold to 1.5 m experimentally.

The associated trajectories are recognized to be the correct player's trajectories, and the average value of the trajectories, in each frame from their start frame to their end frame, is taken as the coordinates of the player in that frame; in addition, the coordinates in successive frames become the track of the player. By placing the number identifying the viewpoint in the correspondence log in the corresponding trajectory, it is made clear which viewpoint has been matched, while avoiding double correspondence with the trajectories of other players. The trajectories for which correspondence has been made are managed collectively using an array data structure. This array is called a track array, and the label numbers of trajectories of each viewpoint that are associated with each player, as shown in Fig. 7, are stored for each frame. Figure 7 shows an example in which the label numbers of the trajectories constituting the trajectories of the 8th to 11th frames are represented. The coordinates of player 1 are calculated from the 1st trajectory of the 1st viewpoint, 101st trajectory of the 2nd viewpoint, and 1001st trajectory of the 3rd viewpoint. The details can be understood by referring to the frame

numbers and trajectory numbers in each trajectory array.

Because there are five basketball players in each team, it is desirable that five tracks exist in each frame. However, there are cases where a trajectory is interrupted because occlusion occurs. We assume that if occlusion occurs from one viewpoint and the trajectory is interrupted, the trajectory continues from other viewpoints. Even if the trajectory is interrupted at a viewpoint where occlusion occurs, tracking can be performed while avoiding the occlusion by correlating the trajectory restored thereafter with the continued trajectory from another viewpoint (see Fig. 8). This is the feature of this method. Therefore, it is necessary to confirm whether the trajectories already associated with each other correspond to the trajectories that have appeared after occlusion in the viewpoint image in which the discontinuation occurs. Even if occlusion occurs, it is desirable that the trajectory of one of the viewpoints is connected, but if occlusion occurs in three or more players at one location, or occlusion occurs continuously in a short time, the trajectory is interrupted from all of the viewpoints. Therefore, exception handling is required in the following cases:

- **One Player Whose Trajectory has been Lost from Three Viewpoints.** The track that has reappeared in the subsequent frame is assigned to this player. In this way, it is possible to avoid mistakes in trajectory assignment.
- **Two or More Players Whose Trajectories have been Lost from three Viewpoints.** Although it would be possible to track several players after the occurrence of an occlusion, it is possible that the player IDs initially assigned to trajectories may be interchanged with one another. This is because it may not be possible to distinguish between players from the same team in the image. If all of the (two or more) trajectories are broken, when the trajectories are restored again and associated with players, each track is assigned to the player whose position is closest to the position of the player that was associated with the track before the break. However, it is difficult to be certain that the track can be reliably reassigned to the correct player. Therefore, the players and tracks that may have been wrongly associated, and the corresponding frame numbers, are managed as a batch. After the processing is completed, track assignments are manually confirmed; if track assignments are confirmed, all such assignments made after that frame are confirmed. This ensures the correctness of tracking.



Figure 9: Result of removing the area outside the court using homography transformation.

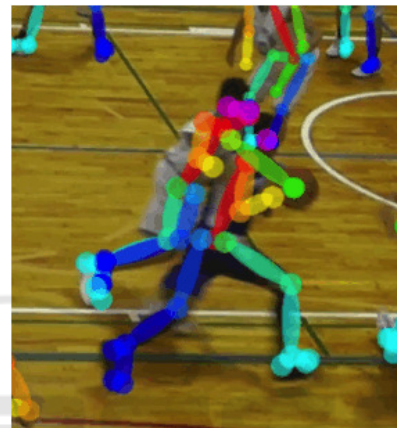


Figure 10: Measurement of human joints by OpenPose.

4 EXPERIMENT

The experiment was performed using videos captured from three viewpoints. The camera used was Panasonic's digital high-definition camera ('Panasonic: HC-V360M, resolution: 1920 x 1080, 30 frames/sec). We selected 180 consecutive frames and tracked players while overcoming the problem of occlusion. Figure 9 shows a processing area in which the area outside the court has been removed using the homography transformation determined during camera calibration.



Figure 11: Multiple players detected using OpenPose.

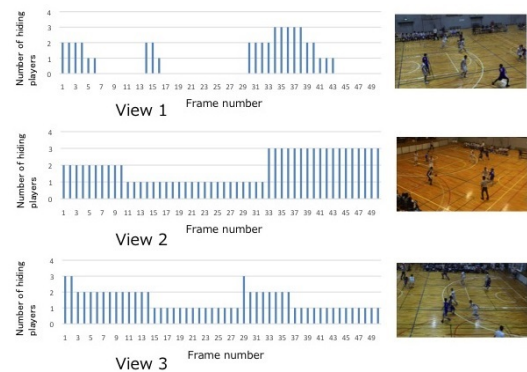


Figure 12: Three views and players' positions on 2-D map using OpenPose.

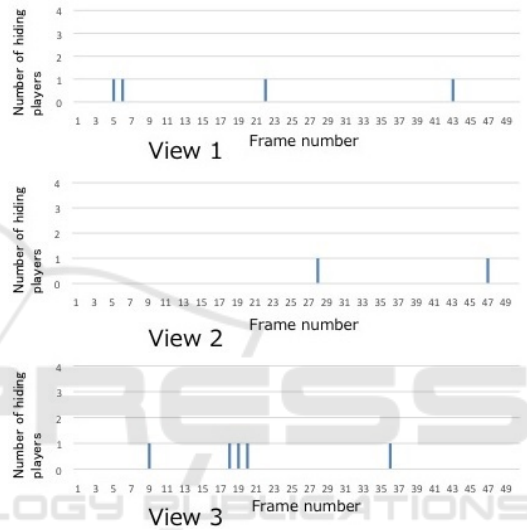
tion.

Figure 10 is an example of joint information obtained by OpenPose, and Fig. 11 shows how OpenPose detects multiple players in an image simultaneously from a certain viewpoint. The appearance of one frame of the player tracking result, when using OpenPose, is shown in Fig. 12, together with the three view images used. Figure 13 shows the number of hidden players in each frame, from each viewpoint. Occlusion is clearly reduced by using OpenPose. When only color information was used, the average occlusion rate per viewpoint was 0.14% for three players, 0.26% for two players, 0.42% for one player, and 0.21% for no occlusion. However, except for one time, occlusion avoidance using multiple viewpoints was performed correctly by the proposed method. The failed case involved frames in which two players could not be detected at the same time from all of three viewpoints, and when the track was recalculated, player substitution occurred. In contrast, when player detection was performed using joint information from OpenPose, occlusion (from more than one viewpoint) did not occur in the same frame. Therefore, players were always detected from at least two viewpoints, and occlusion was avoided in all cases.

Another advantage of using OpenPose is that the resulting trajectories were stable, reducing apparent position errors. With uniform color information, various positions on the back and abdomen were detected as player positions, whereas, when using OpenPose, it was possible to identify and detect the position of the waist with a relatively small difference between players. The star in Fig. 14 indicates the waist position detected by OpenPose from an example image. Therefore, when performing homography transformation, a standard waist height could be used, and the position error on the 2D map was reduced.



(a)



(b)

Figure 13: Time transition of the number of hidden players by method using (a) color and (b) OpenPose.

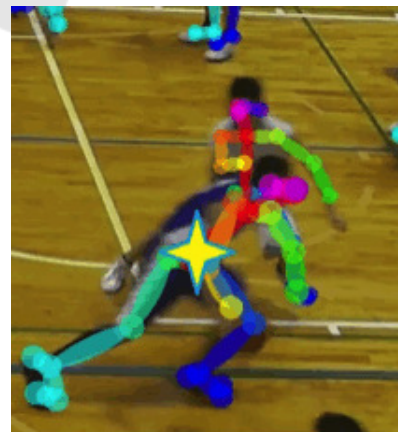


Figure 14: Waist position detected by OpenPose is indicated by a star.

5 CONCLUSIONS

In this study, we proposed a basketball player tracking method that integrates information from multiple viewpoints appropriately. The method is based on video captured by hand-held cameras from around the court and from the spectator seats, and its applicability is very high. We also confirmed that using OpenPose for player detection is very effective, compared with using the team uniform color alone. Because team distinction needs to use uniform color, we plan to extract color information from the OpenPose detection results.

To confirm the effectiveness of integrating information from multiple cameras, we focused on the implementation of algorithms to integrate trajectories obtained from each viewpoint on 2-D reference maps. One of the features of the proposed method is that player tracking at each viewpoint, called trajectory generation, and integration of these trajectories, called trajectory generation, are all performed on the same 2-D reference map using homography. This makes it possible to evaluate the proximity of the detected player position without depending on the position of the player or the camera viewpoint. To operate this algorithm stably, it is necessary to accurately detect trajectories from each viewpoint. Currently, we identify players close to each other in successive frames as the same player, but in the future we plan to add statistical improvements, such as introducing a Kalman Filter (Lu et al., 2013) and Bayesian evaluation (Xing et al., 2011). Building a motion model using the game context (Liu et al., 2013) and modeling the relationship between the ball and the player (Maksai and X. Wang, 2016) are also important issues for accurately tracking the player.

Since joint information by the OpenPose can be used as it is for correspondence from different viewpoints, three-dimensional recognition of joint placement is easy to realize. Therefore, in addition to the closeness of the player position between frames, we are investigating whether the tracking of the player can be made more accurate by using the inter-frame matching of this three-dimensional joint information.

Future issues include three-dimensional recognition of players, application of this method to team play and tactical analysis, and ball detection linked to the recognition of dribbling, passing, and shots. For this purpose, three-dimensional reconstruction from joint information detected by the OpenPose is effective. In recent years, the application of Deep Neural Network that handles time series to human behavior recognition has been actively studied. Application to sports analysis is also underway (Baccouche et al.,

2010), (Tsunoda et al., 2017), (Wang and Zemel, 2016). We plan to develop such a DNN-based method using joint three-dimensional motion information as input.

ACKNOWLEDGEMENTS

We would like to thank Dr. Shinji Ozawa, Emeritus professor of Keio University, Japan, for valuable advice on this research. In addition, we thank Edanz Group (<https://en-author-services.edanzgroup.com/>) for editing a draft of this manuscript. Part of this work was supported by JSPS KAKENHI, grant number 19K12046.

REFERENCES

- Baccouche, M., Mamalet, F., Wolf, C., Garcia, C., and Baskurt, A. (2010). Action classification in soccer videos with long short-term memory recurrent neural networks. In *Proc. Int. Conf. on Artificial Neural Networks*.
- Cao, Z., Simon, T., Wei, S.-E., and Sheikh, Y. (2017). Real-time multi-person 2d pose estimation using part affinity fields. In *Proc. IEEE CVPR*, pages 7291–7299.
- Chen, H.-T., Chou, C.-L., Fu, T.-S., Lee, S.-Y., and Lin, B.-S. P. (2012). Recognizing tactic patterns in broadcast basketball video using player trajectory. *J. Vis. Commun. Image R.*, 23:932–947.
- Chen, H.-T., Tien, M.-C., Chen, Y.-W., Tsai, W.-J., and Lee, S.-Y. (2009). Physics-based ball tracking and 3d trajectory reconstruction with applications to shooting location estimation in basketball video. *J. Vis. Commun. Image R.*, 20:204–216.
- Fu, T.-S., Chen, H.-T., Chou, C.-L., Tsai, W.-J., and Lee, S.-Y. (2011). Screen-strategy analysis in broadcast basketball video using player tracking. In *Proc. IEEE Visual Commun. and Image Process*.
- Hu, M.-C., Chang, M.-H., Wu, J.-L., and Chi, L. (2011). Robust camera calibration and player tracking in broadcast basketball video. *IEEE Trans. Multimedia*, 13(2):266–279.
- Idaka, Y., Yasuda, K., Ho, Y., and Tagawa, N. (2017). Cost-effective camera pose estimation for basketball analysis using radon transform. In *Proc. Int. Conf. Infomatics, Electronics and Vision*.
- Kanatani, K. (1993). *Geometric Computation for Machine Vision*. Oxford University Press, Oxford, U.K.
- Krosshaug, T., Nakamae, A., Boden, B., and et al. (2007). Mechanisms of anterior cruciate ligament injury in basketball: Video analysis of 39 cases. *The American Journal of Sports Medicine*, 35(3):359–367.
- Liu, J., Carr, P., Collins, R.-T., and Liu, Y. (2013). Tracking sports players with context-conditioned motion models. In *Proc. IEEE CVPR*, pages 1830–1837.

- Liu, Y., Liu, X., and Huang, C. (2011). A new method for shot identification in basketball video. *Journal of Software*, 6(8):1468–1475.
- Lu, W.-L., Ting, J.-A., Little, J. J., and Murphy, K. P. (2013). Learning to track and identify players from broadcast sports videos. *IEEE Trans. Pattern Anal. Machine Intell.*, 35(7):1704–1716.
- Lucey, P., Bialkowski, A., Carr, P., Yue, Y., and Matthews, I. (2014). How to get an open shot: Analyzing team movement in basketball using tracking data. In *Proc. MIT SLOAN Sports Analytics Conf.*
- Maksai, A. and X. Wang, P. F. (2016). What players do with the ball: A physically constrained interaction modeling. In *Proc. IEEE CVPR*, pages 972–981.
- Tsunoda, T., Komori, Y., Matsugu, M., and Harada, T. (2017). Football action recognition using hierarchical lstm. In *Proc. IEEE CVPR Workshops*.
- Vaeyens, R., Lenoir, M., Williams, A.-M., Mazyn, L., and Philippaerts, R.-M. (2007). The effects of task constraints on visual search behavior and decision making skill in youth soccer players. *Jour. Sport Exercise Psychology*, 29(2):147–169.
- Wang, K.-C. and Zemel, R. (2016). Classifying nba offensive plays using neural networks. In *Proc. MIT Sloan Sports Analytics Conf.*
- Wen, P.-C., Cheng, W.-C., Wang, Y.-S., Chu, H.-K., Tang, N.-C., and Liao, H.-Y.-M. (2016). Court reconstruction for camera calibration in broadcast basketball videos. *IEEE Trans. Visualization and Computer Graphics*, 22(5):1517–1526.
- Xing, J., Ai, H., Liu, L., and Lao, S. (2011). Multiple player tracking in sports video: A dual-mode two-way bayesian inference approach with progressive observation modeling. *IEEE Trans. Image Processing*, 20(6):1652–1667.
- Xu, M., Orwell, J., and Jones, G. (2004). Tracking football players with multiple cameras. In *Proc. IEEE Int. Conf. Image Processing*, volume 3, pages 2909–2912.