

# Football Players Movement Analysis in Panning Videos

Karol Działowski<sup>[0000–0002–9892–8514]</sup>  
and Paweł Forczmański<sup>[0000–0002–3618–9146]</sup>

West Pomeranian University of Technology, Szczecin, Faculty of Computer Science  
and Information Technology, Żołnierska Str. 49, 71–210 Szczecin, Poland  
karoldzialowski@gmail.com, pforczmanski@wi.zut.edu.pl

**Abstract.** In this paper, we present an end-to-end application to perform automatic multiple player detection, unsupervised labelling, and a semi-automatic approach to finding homographies. We incorporate dense optical flow for modelling camera movement and user-assisted calibration on automatically chosen key-frames. Players detection is performed with a pre-trained YOLOv3 detector and player labelling is done using features in HSV colorspace. The experimental results demonstrate that our method is reliable with generating heatmaps from players' positions in case of moderate camera movement. Major limitations of proposed method are the necessity of manual calibration of characteristic frames, inaccuracy with fast camera movements, and small tolerance of vertical camera movement.

**Keywords:** sports video analysis · soccer player tracking · camera calibration · pitch modelling

## 1 Introduction

Soccer is one of the most popular sport arts watched by millions of people around the world. This popularity led to a growing demand by sports professionals and fans for gathering various data related to the players and the game itself. Recently a lot of research was done in the field of *soccer video analysis*. Such systems can be used for getting insights about whole team or individual player performance, they can support referees in decision making, automatically extract highlights or intelligently control the broadcasting camera [9,16].

Systems for gathering data about team or player performance can reveal aspects that are hidden and not so obvious to the human eye. Such systems can measure the distance covered by players, their speed, average position on the pitch, etc. This data can be used by staff members or professional analysts to improve the team performance or by experts in television [1].

There are many approaches to get accurate time-related positions of the players that change at the ground level. Wearable tracking devices are recently the first choice in collecting such data and are used by the majority of elite teams [17]. Complex camera systems mounted around the stadium are another

option. Multiple fixed cameras can cover the entire pitch, detect the player's position and project them into a virtual top-view image via homography of the ground between the camera image and the top-view image [1,10]. However, those options are complex, expensive and often not affordable for the average lower league team.

Taking into consideration the above facts, there is a demand for an affordable data gathering system that does not require complex infrastructure and sophisticated sensors. We have observed that many teams, from amateur to professional levels, record their performance with a single video camera. Those materials are typically taken from a fixed position with a simple horizontal panning and can be used to measure team/player performance [19,31].

In this paper, we propose a semi-automatic system for tracking players on a soccer pitch using single panning camera. The proposed approach consists of two elements. Firstly, the camera motion is modelled with a dense optical flow which is used for selecting characteristic frames required for user-assisted camera calibration and finding homography. Camera calibration is performed on characteristic frames by manually selecting corresponding points between the camera image and pitch model. It is done once for the whole video stream. Then transformation matrices are found for every camera angle with simple linear interpolation. Then we gather some number of frames from camera input. In each frame, we detect players bounding boxes with YOLOv3 and perform feature extraction by calculating hue and saturation histograms. Unsupervised learning is used for creating a model that discriminates players into five classes.

The second stage is responsible for player detection and player classification. Firstly, the pitch pixels segmentation is done by masking a particular color represented in Hue-Saturation-Value color space, then we use general YOLOv3 detector for getting bounding boxes that describe player positions. We do not train the detector but use network weights from *cvlib* library trained for general-purpose applications [23]. For each player, we perform feature extraction by calculating hue and saturation histograms and classification using the model learned at the previous stage. Each player position is projected into a 2D pitch model with a transformation matrix found at the first stage.

The paper is organized as follows. Related works are presented in Section 2. Section 3 describes our method with focus on pitch segmentation, camera calibration and modelling, players detection, players classification and players position tracking. Experimental results are presented in Section 4, while the conclusions and future research directions are presented in Section 6.

## 2 Related works

### 2.1 Field registration

Systems based on a single camera require pitch registration for representing players' positions in two dimensions. One way of acquiring this relation is by finding the homography placing a camera view into a two-dimensional view assuming the playing surface is flat.

In exemplary approaches [12,24], a user manually calibrates several reference images, then the system calibrates remaining images by finding correspondences from reference images. Recently, fully-automatic methods emerged, as they require no or fewer user interactions [4,28].

Sharma et al. [28] proposed a solution for the registration problem defined as the nearest neighbour search over edge maps and synthetically generated homography pairs. Extracting edge data from input images was done with 3 different approaches: histogram of oriented gradients (HOG) features, chamfer matching, and convolution neural networks (CNN).

Chen and Little [4] improved the idea of using synthetic data incorporating Sharma et al. work with generative adversarial network (GAN) model for detecting field marking and siamese networks for finding closest pair of synthetic edge map and homography.

## 2.2 Player detection and reidentification

Player detection can be done in multiple ways. Santhosh and Kaarthick [27] researched the use of HOG color-based detector. Johnson [15] based his work on an open-source multiperson pose estimator named Alpha Pose. Using CNN-base multibox detector for player detection was researched by Ramanathan et al. [25].

Another important aspect is player classification into five main classes corresponding to two teams, two goalkeepers, and referee, namely, player labelling which is usually done with color features [20].

Player tracking (reidentification) solves the problem of temporal correspondence. This step is required for tracking consistent trajectories for each player. A large number of tracking algorithms were used for solving this problem, such as Kalman filter [21], particle filter [1], mean-shift [6], etc.

## 3 Method description

In our case, we use a panning camera which means swivelling a video camera horizontally from a fixed position. This motion is similar to the motion of a person when he/she turns his/her head on a neck from the left to the right.

The term panning is derived from panorama, suggesting an expansive view that exceeds the gaze, forcing the viewers to turn their heads in order to take everything in. Panning, in other words, is a device for gradually revealing and incorporating off-screen space into the image.

In this case, panning refers to the horizontal scrolling of an image wider than the display.

Proposed solution works only in an offline mode as camera motion modelling is required step in the process. This drawback could be solved using more advanced techniques as explained in section 5.

The system is composed of two stages. In the first stage, we model camera angle changes using dense optical flow. Then semi-automatic camera calibration is performed and the team classification model is learned on few examples. The

second stage is fully automatic and consists of pitch detection, player detection, players team classification, and projection of positions from image coordinates to pitch coordinates.

### 3.1 Pitch segmentation

Pitch segmentation is commonly used to eliminate spectator regions and reduce false alarms in player detection. A pitch mask is used in the camera angle modelling process using dense optical flow. Pitch segmentation is performed by selecting color ranges in the HSV color scheme.

Color range was chosen empirically and is defined as a pair of lower bound  $L = (35, 50, 50)$  and upper bound  $U = (75, 255, 255)$ . This creates a problem when pitch color is not in defined color range or non-pitch areas are also green. This problem can be solved with other methods, e.g. segmentation GAN [4].

There are objects such as players, referees, ball and lines on the pitch. To eliminate those objects from the mask we use morphological operations, e.g. closing operation which consists of dilatation and erosion. Figure 1 shows the process of pitch segmentation.

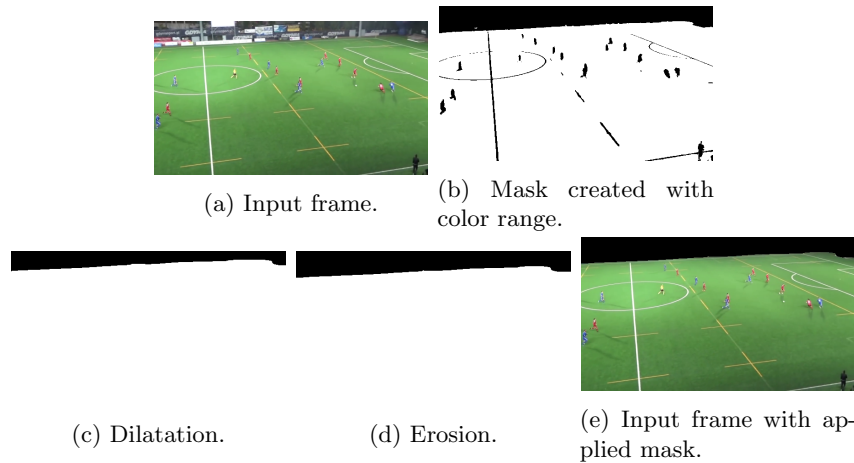


Fig. 1: Process of pitch segmentation.

### 3.2 Camera angle modelling

Analysis of camera angle is accomplished with the use of dense optical flow (calculated for every pixel in the image). It describes changes between the subsequent frames which are a result of moving objects or change of camera parameters [2]. Optical flow works well if pixel intensities of moving object do not change between consecutive frames and neighbouring pixels have similar motion. These

assumptions are met in case of football field observation. The classical equation of optical flow is defined as:

$$I(x, y, t) - I(x + \Delta x, y + \Delta y, t + \Delta t) = 0, \quad (1)$$

where pixel with intensity  $I(x, y, t)$  has moved by  $\Delta x$ ,  $\Delta y$  and  $\Delta t$  between two image frames [8].

After finding the pitch mask we inverse it and calculate optical flow for each frame. This allows us to reduce noise generated by objects moving on the pitch, i.e. players, referee and ball. For every frame, we calculate the mean optical flow vector which describes camera movement in a given frame. The accumulated sum of those means approximates camera angle in relation to the first frame. Example values for footage of panning to the left and then to the right are shown in Fig. 2. Based on this analysis we can choose characteristic key frames with the maximum and minimum value of the camera swing angle in comparison to the first frame of the input video.

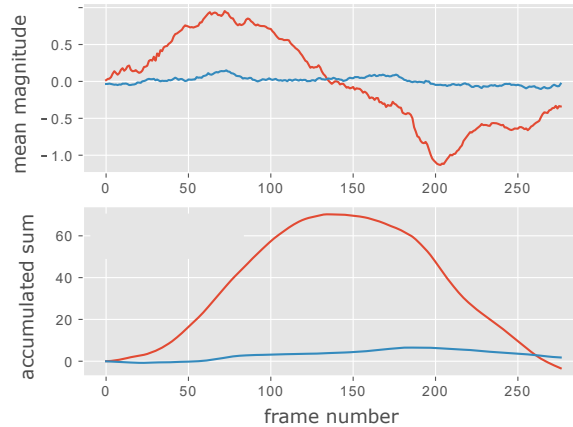


Fig. 2: Camera angle approximation with optical flow. Red line represents horizontal (panning) motion, blue line represents vertical motion. Accumulated sum represents camera angle in given frame compared to first frame of input video.

### 3.3 Camera calibration

Characteristic key frames are used for projective transformation. It allows for a mapping of one plane to another. In our algorithm, we use it to find ground position of players seen in different camera views [22]. Projecting transformation

is expressed by the equation:

$$\begin{bmatrix} x_1 \\ y_1 \\ 1 \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \begin{bmatrix} x_2 \\ y_2 \\ 1 \end{bmatrix}, \quad (2)$$

where  $x_1$  and  $y_1$  are the coordinates of a single point on the input plane,  $x_2$  and  $y_2$  are the coordinates on the output plane,  $H$  is a transformation matrix.

In order to calculate  $H$  by means of least squares method we need to collect four pairs of so called calibration points. Such an approach is a compromise between computational complexity and the quality of the resulting transformation. Exemplary calibration points are presented in Fig. 3.

Knowing transformation matrices  $H_n$  and  $H_m$  corresponding to frames with maximum and minimum horizontal angle, respectively, we can interpolate transformation matrix for each frame using values from maximum – minimum range, as follows:

$$H_k = H_n + \frac{k(H_m - H_n)}{(m - n)}, \quad (3)$$

where  $k$  is camera angle, and  $n < k < m$ .

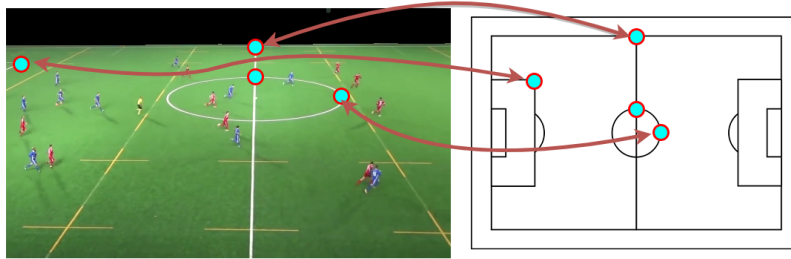


Fig. 3: Calibration points

### 3.4 Players detection

Players detection is performed by the YOLOv3 detector. This was a state-of-the-art detection method at the time of developing our method. The YOLO (You Only Look Once) algorithm proposed by Joseph Redmon and Ross Girshick solves object detection as a regression problem and outputs the location and class of an input object on an end-to-end network in one step [26,30].

We perform detection using the YOLOv3 model trained on the COCO dataset capable of detecting 80 common objects [18] (e.g. cars, cats, dogs, pedestrians, etc.). We used detector *as is* with pre-trained weights from *cvlib* library [23].



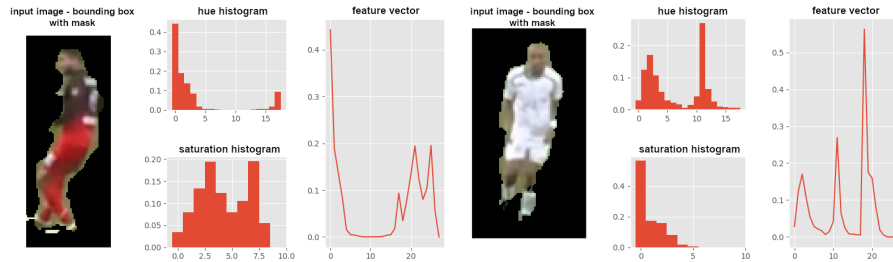


Fig. 5: Feature extraction for team assignment based on hue and saturation histograms.

After team assignment each player is given unique identifier based on identification step in the previous frame, i.e. we give the same identifier to the player with closest spatial distance from the preceding frame.

Having this information we apply post-processing for team assignment step. Dominant team label from previous 11 frames is accepted as a valid label in the current frame. This approach reduces false alarms in the team assignment step.

## 4 Experimental results

### 4.1 Experimental environment

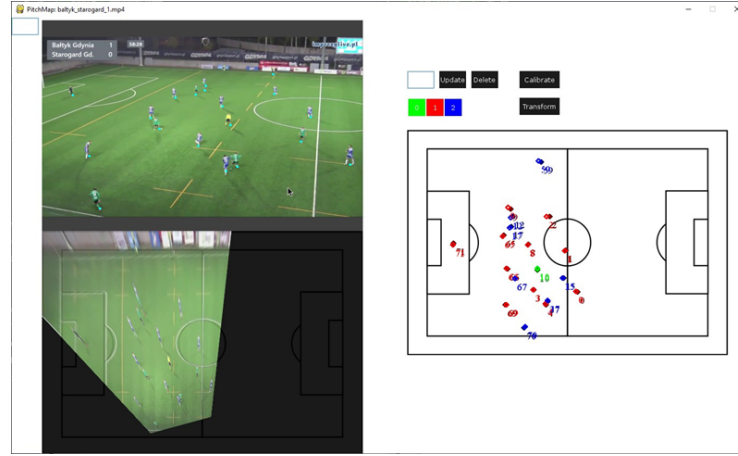
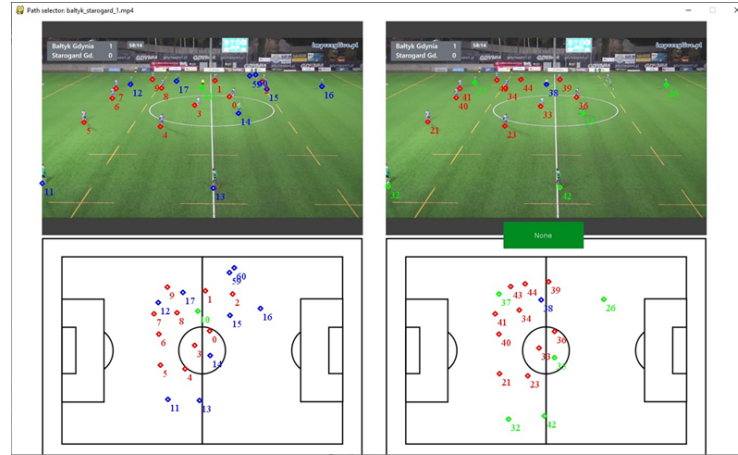
We prepared several separate applications for the experimental part of the work:

- *Pitchmap* - main program, which performs the entire process of calibration, detection and projection on the pitch map,
- *Annotator* - a program for manual calibration and marking of players in the input material, which are later used for comparison with our algorithm,
- *Comparator* - a program for comparing positions from *Pitchmap* and positions from *Annotator*,
- *Heatmap* - a program for generating heatmaps and path comparisons.

The output from *Heatmap* application is shown in Fig. 8a. and Fig. 8b. It contains heatmap generated from 18-seconds-long footage and single player trajectory, respectively.

The experimental protocol is as follows. Firstly, we annotate benchmark videos presenting short fragments of football matches (7-30 seconds long). Two videos have been recorded during amateur league match, while one video material comes from a television broadcast. The first video is rather steady and contains eight changes of a viewpoint. The camera in the second one has higher dynamics, it slowly moves towards right pitch side and then returns with a significantly higher speed. The last material contains continuous camera movement in one direction, after some time, the camera returns to its initial position. What is a little problematic, the camera's viewpoint moves slightly up and down.



Fig. 6: Main window of *Annotator* application.Fig. 7: Main window of *Comparator* application.

During experiments we calculated paths of moving players and the heatmaps of their presence on the pitch. Finally, we compared the results of automatic procedure with manual annotations by means of objective Structural SIMilarity and mean distance between positions of individual players.

## 4.2 Results

The main focus has been put on the calibration procedure, since it has the greatest influence on the algorithm results. The following calibration methods has been be compared:

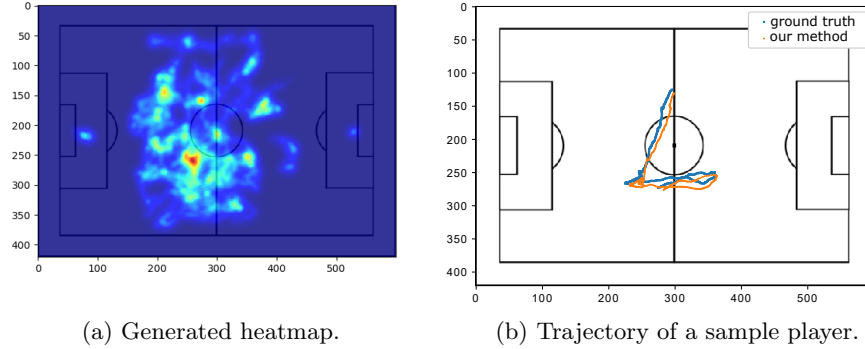


Fig.8: Exemplary output generated based on data aggregated by proposed method from 18 seconds footage with horizontal panning.

1. Calibration with three characteristic frames - calibration is performed on two frames with the greatest camera inclination and on a frame with central position. All intermediate tilt angles are interpolated.
2. Calibration with two characteristic frames - calibration is performed on the two frames with the greatest camera inclination. All intermediate tilt angles are interpolated.
3. Calibration with manual feature frames - the user selects two key frames. All intermediate frames are interpolated.

The evaluation was performed using two methods: by means of image similarity comparison between heatmaps and by means of individual player path comparison. The image similarity was estimated using standard Structural Similarity (SSIM) method [29], assuming the perfect match is represented by value close to one. The paths were compared using mean distance between vectors  $P$  and  $Q$  (of  $n$  frames) representing annotated (ground truth) position and estimated one, respectively:

$$d = \frac{1}{n} \sum_{i=0}^{n-1} \|(P_i - Q_i)\|^2. \quad (4)$$

The results of heatmap comparison are presented in Tab. 1. It contains values of SSIM for three benchmark videos and two calibration methods. For the comparison purpose, the result of manual calibration was also given, yet it should be noted that it was estimated for a reduced material time-span (due to complex process of manual annotation of longer video sequences).

As it was observed, SSIM can not always be mapped onto a subjective assessment of similarity. The results show also that the calibration with three characteristic frames usually gives better results than calibration with two characteristic frames.

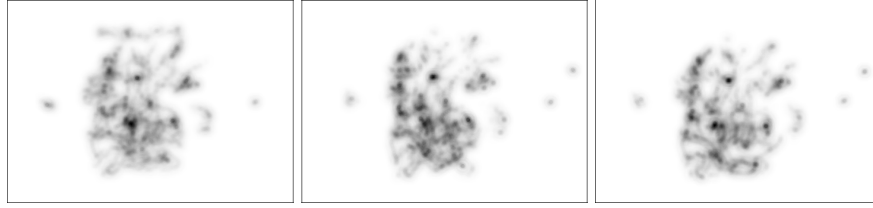


Fig. 9: Exemplary heatmaps for video nr.1: Manually annotated (left), semi-automatic with three key-frames (middle), semi-automatic with two key-frames (right).

Table 1: SSIM values for heatmaps generated using different calibration methods

calibration method	video material		
	1	2	3
three key frames	0.875	0.871	0.831
two key frames	0.857	0.858	0.834
manual (for reduced time span)	0.866	0.852	0.904

The results of paths comparison are presented in Tab. 2. As in case of SSIM, it contains values of distance for three benchmark videos and two calibration methods. For the comparison purpose, the result of manual calibration was also given, yet it should be noted that it was estimated for a reduced material time-span (due to complex process of manual annotation of longer video sequences).

Figure 10 presents a projection onto X and Y axis for player path, for two calibration methods. Subjectively, the method with two key frames gives the better results, however the closer look at the objective measures (see Tab. 2) unveils that the method with three key frames is better.

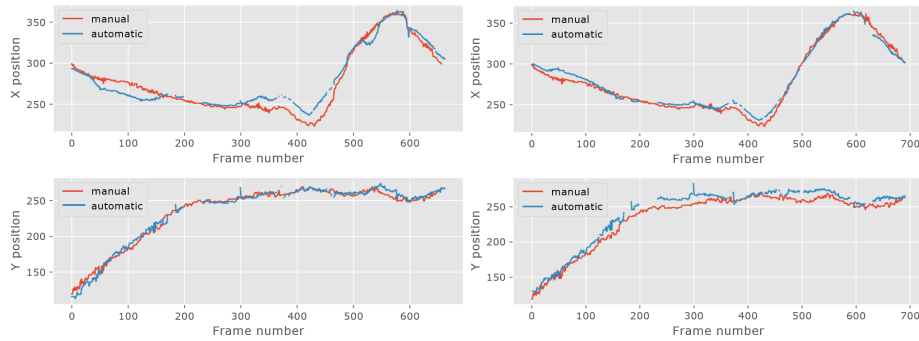


Fig. 10: Exemplary paths for individual player for video nr.1: semi-automatic calibration with three key-frames (left), semi-automatic with two key-frames (right).

Table 2: Mean distances (in pixels) between paths travelled by selected player and different calibration methods

calibration method	video material		
	1	2	3
three key frames	9.694	9.816	27.302
two key frames	9.908	13.399	28.444
manual (for reduced time span)	18.063	10.562	9.438

## 5 Limitations and future work

Our proof-of-concept work can be improved in some areas. Replacing the manual calibration step with a fully automatic one [4,5,13] could significantly improve usability and allow on-site analysis. Manual calibration step can also reduce calibration accuracy in comparison to automatic solutions.

The second limitation is input constraints. Our method accepts only continuous footage with a horizontal pan so it cannot be applied on raw broadcast footage. However, this can be improved with automatic scene detection [14].

It should be noted that the proposed system is not robust in all situations. We have observed major errors when the camera angle changes frequently with significant speed. The proposed solution has a small tolerance for vertical panning movement which imposes a requirement for only horizontal panning footage. Those problems can be solved by slicing input video (e.g. by tensor-based methods [7]) or incorporating more sophisticated methods of camera calibration (e.g. [3]).

Another limitation is the requirement that the football pitch is green. The color range used for segmentation was chosen arbitrarily so proposed system is not universal for all kinds of surfaces (e.g. snow conditions, artificial surface in other colors, indoor hard courts), however the color range can be easily tweaked if necessary or other methods of segmentation could be used.

## 6 Conclusions

Knowing players position on the pitch we can perform analysis tasks like heatmap generation or players trajectory analysis. Experimental results show usefulness of the proposed method for player tracking, which uses homography and integrates the player information from single camera on the virtual ground image.

By having players positions in the soccer scene it could be used in some applications such as team strategy analysis, scene recovery and measuring individual players performance.

## References

1. Baysal, S., Duygulu, P.: Sentioscope: a soccer player tracking system using model field particles. *IEEE Transactions on Circuits and Systems for Video Technology* **26**(7), 1350–1362 (2015)
2. Beauchemin, S.S., Barron, J.L.: The computation of optical flow. *ACM computing surveys (CSUR)* **27**(3), 433–466 (1995)
3. Chen, J., Little, J.J.: Sports camera calibration via synthetic data. *CoRR* **abs/1810.10658** (2018), <http://arxiv.org/abs/1810.10658>
4. Chen, J., Little, J.J.: Sports camera calibration via synthetic data. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*. pp. 0–0 (2019)
5. Chen, J., Zhu, F., Little, J.J.: A two-point method for ptz camera calibration in sports. In: *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*. pp. 287–295. IEEE (2018)
6. Chiang, T.K., Leou, J.J., Lin, C.S.: An improved mean shift algorithm based tracking system for soccer game analysis. In: *Proceedings: APSIPA ASC 2009: Asia-Pacific Signal and Information Processing Association, 2009 Annual Summit and Conference*. pp. 380–385. Asia-Pacific Signal and Information Processing Association, 2009 Annual ... (2009)
7. Cyganek, B., Woźniak, M.: Tensor-based shot boundary detection in video streams. *New Generation Computing* **35**(4), 311–340 (Oct 2017). <https://doi.org/10.1007/s00354-017-0024-0>, <https://doi.org/10.1007/s00354-017-0024-0>
8. Dalka, P.: Methods of algorithmic analysis of the video image for applications in traffic monitoring [in Polish: Metody algorytmicznej analizy obrazu wizyjnego do zastosowań w monitorowaniu ruchu drogowego]. Ph.D. thesis, Gdansk University of Technology (2015)
9. D’Orazio, T., Leo, M.: A review of vision-based systems for soccer video analysis. *Pattern recognition* **43**(8), 2911–2926 (2010)
10. Enomoto, A., Saito, H.: Ar display for observing sports events based on camera tracking using pattern of ground. In: *International Conference on Virtual and Mixed Reality*. pp. 421–430. Springer (2009)
11. Frejlichowski, D.: A method for data extraction from video sequences for automatic identification of football players based on their numbers. In: Maino, G., Foresti, G.L. (eds.) *Image Analysis and Processing – ICIAP 2011*. pp. 356–364. Springer Berlin Heidelberg, Berlin, Heidelberg (2011)
12. Ghanem, B., Zhang, T., Ahuja, N.: Robust video registration applied to field-sports video analysis. In: *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*. vol. 2. Citeseer (2012)
13. Homayounfar, N., Fidler, S., Urtasun, R.: Sports field localization via deep structured models. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. pp. 5212–5220 (2017)
14. Jinjun Wang, Engsiong Chng, Changsheng Xu: Soccer replay detection using scene transition structure analysis. In: *Proceedings. (ICASSP ’05). IEEE International Conference on Acoustics, Speech, and Signal Processing, 2005*. vol. 2, pp. ii/433–ii/436 Vol. 2 (2005). <https://doi.org/10.1109/ICASSP.2005.1415434>
15. Johnson, N.: Extracting player tracking data from video using non-stationary cameras and a combination of computer vision techniques. In: *Proceedings of the 14th MIT Sloan Sports Analytics Conference, Boston, MA, USA* (2020)

16. Larson, N.G., Stevens, K.A.: Automated camera-based tracking system for sports contests (Nov 8 1994), uS Patent 5,363,297
17. Leveaux, R., Messerschmitt, M.: The changing shape of sport through information technologies. In: Proceedings of the 26th International Business Information Management Association Conference-Innovation Management and Sustainable Economic Competitive Advantage: From Regional Development to Global Growth, IBIMA 2015 (2015)
18. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: European conference on computer vision. pp. 740–755. Springer (2014)
19. Mackowiak, S., Konieczny, J.: Player extraction in sports video sequences. In: 2012 19th International Conference on Systems, Signals and Image Processing (IWSSIP). pp. 409–412. IEEE (2012)
20. Manafifard, M., Ebadi, H., Moghaddam, H.A.: A survey on player tracking in soccer videos. *Computer Vision and Image Understanding* **159**, 19–46 (2017)
21. Najafzadeh, N., Fotouhi, M., Kasaei, S.: Multiple soccer players tracking. In: 2015 The International Symposium on Artificial Intelligence and Signal Processing (AISP). pp. 310–315. IEEE (2015)
22. Nowosielski, A., Frejlichowski, D., Forczmański, P., Gosciewska, K., Hofman, R.: Automatic Analysis of Vehicle Trajectory Applied to Visual Surveillance. In: Choras, RS (ed.) *IMAGE PROCESSING AND COMMUNICATIONS CHALLENGES 7. Advances in Intelligent Systems and Computing*, vol. 389, pp. 89–96 (2016). [https://doi.org/10.1007/978-3-319-23814-2\\_11](https://doi.org/10.1007/978-3-319-23814-2_11)
23. Ponnusamy, A.: cvlib - high level computer vision library for python. <https://github.com/arunponnusamy/cvlib> (2018)
24. Puwein, J., Ziegler, R., Vogel, J., Pollefeys, M.: Robust multi-view camera calibration for wide-baseline camera networks. In: 2011 IEEE Workshop on Applications of Computer Vision (WACV). pp. 321–328. IEEE (2011)
25. Ramanathan, V., Huang, J., Abu-El-Haija, S., Gorban, A., Murphy, K., Fei-Fei, L.: Detecting events and key actors in multi-person videos. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 3043–3053 (2016)
26. Redmon, J., Farhadi, A.: Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767* (2018)
27. Santhosh, P., Kaarthick, B.: An automated player detection and tracking in basketball game. *Computers, Materials & Continua* **58**(3), 625–639 (2019)
28. Sharma, R.A., Bhat, B., Gandhi, V., Jawahar, C.: Automated top view registration of broadcast football videos. In: 2018 IEEE Winter Conference on Applications of Computer Vision (WACV). pp. 305–313. IEEE (2018)
29. Wang, Z., Simoncelli, E.P., Bovik, A.C.: Multiscale structural similarity for image quality assessment. In: The Thirty-Seventh Asilomar Conference on Signals, Systems Computers, 2003. vol. 2, pp. 1398–1402 Vol.2 (2003). <https://doi.org/10.1109/ACSSC.2003.1292216>
30. Yi, Z., Yongliang, S., Jun, Z.: An improved tiny-yolov3 pedestrian detection algorithm. *Optik* **183**, 17–23 (2019)
31. Zhu, G., Huang, Q., Xu, C., Rui, Y., Jiang, S., Gao, W., Yao, H.: Trajectory based event tactics analysis in broadcast sports video. In: Proceedings of the 15th ACM international conference on Multimedia. pp. 58–67 (2007)