Identifying Basketball Plays from Sensor Data; towards a Low-Cost Automatic Extraction of Advanced Statistics

Sangüesa, Adrià Arbués; Moeslund, Thomas B.; Bahnsen, Chris Holmberg; Iglesias, Raul Benítez

Published in:
2017 IEEE International Conference on Data Mining Workshops (ICDMW)

DOI (link to publication from Publisher):
10.1109/ICDMW.2017.123

Publication date:
2017

Document Version
Accepted author manuscript, peer reviewed version

Identifying Basketball Plays from Sensor Data; towards a Low-Cost Automatic Extraction of Advanced Statistics

Adrià Arbués Sangüesa
Aalborg University
Aalborg, Denmark
adria.arbues@upf.edu

Thomas B. Moeslund
Aalborg University
Aalborg, Denmark
tbm@create.aau.dk

Chris H. Bahnsen
Aalborg University
Aalborg, Denmark
cb@create.aau.dk

Raul Benítez Iglesias
Polytechnic University of Catalonia
Barcelona, Spain
raul.benitez@upc.edu

***Abstract*—Advanced statistics have proved to be a crucial tool for basketball coaches in order to improve training skills. Indeed, the performance of the team can be further optimized by studying the behaviour of players under certain conditions. In the United States of America, companies such as STATS or Second Spectrum use a complex multi-camera setup to deliver advanced statistics to all NBA teams, but the price of this service is far beyond the budget of the vast majority of European teams. For this reason, a first prototype based on positioning sensors is presented. An experimental dataset has been created and meaningful basketball features have been extracted. 97.9% accuracy is obtained using Support Vector Machines when identifying 5 different classic plays: floppy offense, pick and roll, press break, post-up situation and fast breaks. After recognizing these plays in video sequences, advanced statistics could be extracted with ease.**

***Index Terms*—Acceleration Wearable Sensors, Basketball, Player Tracking, Play Classification, Advanced Statistics**

I. Introduction

In February 2016, the National Basketball Association (NBA) extended their existing deal with North-American televisions (ESPN and TNT), which meant a notable increase in team salary caps [3]. This fact encouraged many general managers to invest a bigger part of the budget in technology. Their goal was clear enough: with a deep analysis of existing team data, resources can be optimized in order to win more games. This analysis can be done by using different advanced statistics. An example could be the number of points per game a certain player scores after executing a specific play in road games. Data is a powerful tool for coaches and can be used to identify team's strengths and weaknesses, to scout another team/a certain player or even to prevent injuries.

Within this framework, two companies offering advanced statistics services have recently emerged: STATS (more concretely, Sports VU) [14] and Second Spectrum [13]. Their products are based on a multi-camera configuration system (in the case of STATS, 6 cameras are used), and both companies manage to track all players and the ball at 25 frames per second. It is clear that this service was appealing to both NBA teams and to the league itself: STATS became the official statistics distribution partner and every single stadium has their camera setup installed. Besides, Second Spectrum is the official tracking provider. However, this model is not being used in Europe for a simple reason: the budget. The lowest salary cap of a NBA team is 79 million dollars [5], whilst the biggest among European teams is not above 38 millions. As the technological solutions offered by STATS and Second Spectrum are expensive, alternatives must be found in order to make advanced statistics available to (at least) top European teams.

A cheaper technological approach would imply the use of positioning sensors like the ones commercialized by the Spanish company Nothing But Net 23 (NBN23) [10]. Such wearable sensors can be conveniently placed in the player's shorts lace or even in their trainers. With these sensors, coaches receive physical data of their players, such as speed or acceleration, and visual statistics like heat maps. Nevertheless, technical and tactical details of the game are not being currently extracted.

The goal of this project is to enrich sensors' data with basketball knowledge by understanding which plays are occurring on court, in order to build a low-cost solution that could provide European teams with similar benefits to the ones the NBA has. For the presented test, NBN23 positioning sensors are used to track the players, and a new approach to manually tag the ball in *real-time* is designed; having integrated both ball and players' data, different basketball-meaningful features are extracted for each play, thus training a model capable to successfully distinguish between four classic basketball plays: *floppy offense*, pick and roll, post-up situation and press break (a brief playbook can be found in Appendix A).

The rest of this paper is organized as follows: in Section II, the state of the art is reviewed, in Section III, the proposed system is detailed; results are shown in Section IV and discussed in Section V; finally, conclusions are extracted in Section VI.

II. Related Work

Although the field that mixes basketball with Machine Learning and Computer Vision techniques is not the most explored one, several contributions have been made.

In the paper written by Wang and Zemel [16], an algorithm to classify a closed-set of plays using Recurrent Neural Networks (RNN) is presented, with the purpose of generating detailed reports with a high-level basketball understanding. Having Sports VU raw tracking data of NBA players, they

turn the problem into image classification by transforming coordinates into a pictorial representation. Besides, the positions of the players are guessed by comparing their shooting tendencies. Their results seem to be promising, but the system is thought for a particular team in a specific season, so it is not automatically tuned to any kind of team.

Using the same Sports VU raw data, Lucey *et al.* [8] analysed how teams manage to have open shots in order to improve shooting percentages. First, their algorithm assigns a role (position) to every player at the beginning of the action. Then, the different factors that may affect when attempting a shot are checked from a more analytic point of view. Finally, different plays are retrieved using tracking data, which clusters similar plays into permutations from the original one. Extracted results show that one of the most relevant features is the defending swaps that may occur during the game, also called mismatches. Although it is rather a statistics paper, this project aims to the same goal: to extract relevant information from tracking data to improve the understanding of the game.

Miller and Bornn [9] made another relevant contribution. They organized a large set of plays by grouping structural similarities, as they observed that there is not an efficient scouting method for professional basketball teams. Their goal is achieved through a segmentation of short plays to shorter manageable segments, a possession modelling by adapting topic models and a bag-of-words structure. Finally, having clustered data with nearest-neighbours algorithms, different types of analysis are done. Although the attached videos show promising results, no numerical results are displayed.

Another article about advanced statistics was presented by Felsen and Lucey [4]. In their study, the goal was to find correlations between different types of shots and the body position of the shooter. Their motivation was to complement the Sports VU data, because taking only coordinates into account, some relevant information may be missed. Their method includes a quantification of the involved anatomy in a three-point-shot and a machine learning module, where a model is trained to identify open/tough shots and to attribute correlations by comparing open against tough shots, and made against missed shots. Furthermore, the authors also performed a deep analysis of the shooting parameters of the best NBA shooter (Stephen Curry), and found out out that, although there are many biometric correlated factors in open/tough shots, those cannot be generalized into a single model.

Ramanathan *et al.* [12] published a method to recognize event and key actors in multi-person videos by detecting the focus of attention of different basketball plays. The goal of this research was to amend the lack of a a universal method to emphasize attention or include key actors in sport sequences. In order to carry out this project, they manually labelled sets of plays of Youtube basketball games. Then, for every class, they extracted features including both scene and particular player information; then, a deep learning framework is used to classify. To properly track the players, the Lucas-Kanade tracker [1] is implemented in combination with a bipartite graph, which is used for matching. Their event detection method is done through a sliding window technique that displays attention with a heat-map. Their results are outperform some state of the art methods, and their dataset can be found on-line. However, their number of classes is simplified to few similar plays, and their tracking system is based on positions in the screen, and not real coordinates in the basketball court.

Another approach to track basketball players through video processing and perform data analysis was thought by Perse *et al.* [2], [11]. With a 2-camera configuration setup in the ceiling of the arena, a method could be designed in order to help planning training sessions based on players' movements. Their method creates a play-designer module, which contains a playbook of stored templates with different plays. Then, the phase of the game (offensive / defensive / time-out) is found by clustering the distribution of players on court with a Gaussian Mixture Model [15]. Afterwards, the small-scale parts of the game are found: the court is divided into 9 sections and basic events are used in order to define the player motion on the court. Finally, recognition is done by using the stored templates in the play-designer. Although their dataset was not huge, their results are consistent; nevertheless, there is no ball information and the algorithm does not have the possibility of learning new plays on its own.

III. Proposed System

In this Section, the whole system and algorithms are detailed. First, the gathered dataset is described together with a brief discussion of the pros and cons of using positioning sensors. Then, a new approach to accurately track the ball almost in real-time based on manual tags and hotkeys is presented, and its integration with players' data is explained as well. Finally, the *3-player attention* approach and the feature extraction process are specified.

A. Experimental Dataset

Having seen the limitations of the on-line existing annotated datasets (coordinates relative to the screen, short video sequences), a new one was created from scratch. As mentioned in Section I, the idea is to build a low-cost system, so positioning sensors (borrowed from the company NBN23) were used instead of a multi-camera configuration setup. The dataset contains 30 minutes of a whole practice of the Under-21 Team of *Valencia Basket Club* (Spanish team, in Valencia), with all 10 players and a coach wearing sensors. More concretely, the most relevant content of this practice can be divided (by a basketball expert) into 96 observations or drills:

- 22 repetitions of a 3-on-0 (3 offensive players, 0 defensive) exercise to practice *floppy offense* motion.
- 22 repetitions of a 4-on-0 exercise starting with a pick and roll.
- 14 repetitions of a 3-on-3 press break exercise to overcome defensive pressure.

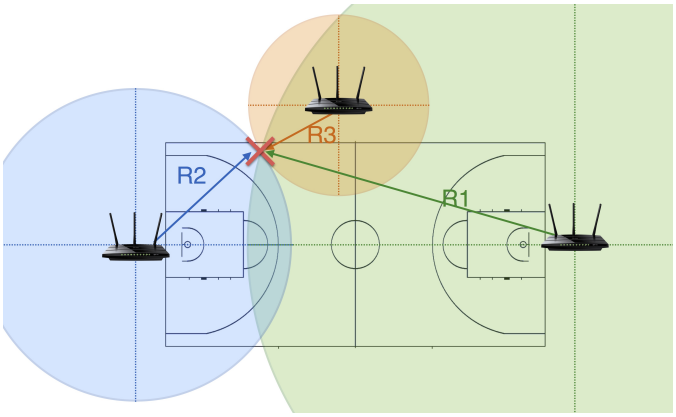

Fig. 1. Signal triangulation through amplitude signals; with three receptors, the exact position of a player inside the court (red cross) is obtained.

- 21 repetitions of a 3-on-2 post-up exercise.
- 17 repetitions of a fast break exercise.

NOTE: all these plays are explained in Appendix A.

Despite working with sensors data, the practice was also recorded with a single static camera (neither with panning nor zooming). The reason for doing so will be described in detail in Section III-B. This dataset (video plus tracking data) cannot be found online, as it belongs to NBN23.

1) Accelerometric Weareble Sensors: Accelerometric NBN23 sensors emit amplitude Bluetooth signals at a frame rate of 25 fps, which are then captured by 3 receptors placed in the court at pre-established spatial locations. Real-time receptors send the captured information to a server, and a script creates an individual *.csv* file for each player containing all his/her corresponding data. By triangulating the signals (as seen in Figure 1), the emission can be decoded in order to obtain the following information: Timestamp, ID and X and Y position in the court (measured in meters).

On the one hand, the two main drawbacks of working with sensors are evident: (a) data can only be extracted in those teams that use sensors, so the option of scouting another team is *a priori* discarded, and (b) the ball also needs to have an integrated sensor; otherwise, it must be tracked somehow. On the other hand, sensors are an easy and cheap technology to be used in team practices, as those do not require *extra-employees* in court; when using cameras, people in charge of recording and monitoring the audio-visual devices are needed.

B. Tracking the Ball using manual tags

In the presented experiment, it was not possible to integrate a chip to all the balls the Valencian club had, so an alternative had to be found. The solution was based on the following principle: *you can estimate the ball position even if you do not know the exact coordinates; you just need to know which player has it*. For this reason, video tracking techniques are discarded and a simpler procedure is chosen. Having recorded the game/practice, a program with hotkeys is designed to create another *.csv* file, containing annotations with the current frame and a tag indicating the type of action, which could be one of the following:

- A player gets the ball (receives from another player / grabs a rebound / steals it).
- A player releases the ball (passes to another player / attempts a shot / looses it).
- A player is substituted (starts playing / goes to the bench).
- The ball touches the rim.
- The game is paused/resumed.

Besides, from the lecture of these tags, the following set of statistical features is extracted: *e.g.* if a tag *N* says that *player A* releases the ball and the tag *N+1* indicates that the ball touches the rim, it is obvious that *player A* attempted a shot; otherwise, if the *N+1* tag indicates that another player receives the ball, that action was a pass. Moreover, advanced parameters such as the speed of pass can be estimated too.

The final hotkey configuration used in this experiment can be seen in Figure 2; as it can be observed, one same key is used to indicate if a player gets or releases the ball, as it is a binary state, and the same happens with substitutions. Another positive consequence of labelling actions with this approach is that different repetitions of an exercise can be temporally divided because of stop/resume tags. This fact simplifies the feature extraction of individual plays, as it will be seen in Section III-E.

In terms of speed, annotations can almost be generated in real-time; different tests were performed and it was estimated that the time to tag a video of duration T is $1.15 \times T$.

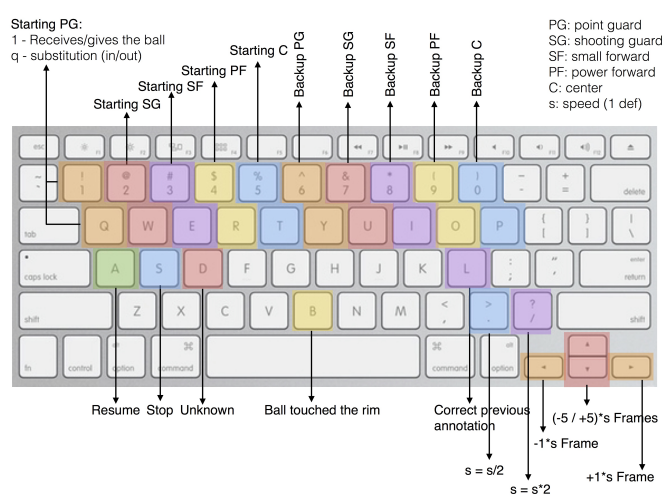

Fig. 2. Final hotkey configuration for labelling ball events. Note that PG, SG, SF, PF and C are the different basketball positions.

C. Parsing Data into a 2D Representation

Once the information of both ball and players was obtained in different files, data had to be synchronized and merged together in a single matrix sorting by timestamp values. It has to be taken into account that sensors start emitting when the player activates them, so there is not a universal beginning

for all signals. Inside this matrix, all the samples that can be comprised into time intervals of 40 milliseconds will correspond to the same frame.

Having the tracking data of the whole sequence, an animated 2D pictorial representation can be generated over a court image in order to have a visual support of the practice/game without occlusions; as it can be observed in Figure 3, every square represents a player and the sensor ID determines its colour. The only thing to be considered is that the decodification of the sensor signal takes as a reference the centre of the court and it is horizontally flipped with respect to the camera point of view, so a conversion has to be applied in order to obtain the player position in pixels. Having the following variables:

$$\text{im}_{\text{size}} = (\text{im}_{\text{width}}, \text{im}_{\text{height}}) \quad [\text{pixels}]$$

$$\text{court}_{\text{size}} = (28, 15) \quad [\text{meters}]$$

$$\text{im}_{\text{center}} = (\tfrac{\text{im}_{\text{width}}}{2}, \tfrac{\text{im}_{\text{height}}}{2}) = (h_x, h_y)$$

$$f_W = \tfrac{\text{im}_{\text{width}}}{28}; f_H = \tfrac{\text{im}_{\text{height}}}{15}$$

given a point (x, y) in meters representing the player location inside the real court, the mapped point (X, Y) in the image expressed in pixels is:

$$(X, Y) = \begin{cases} (h_x - (x * f_W), h_y + (x * f_H)) & \text{if } (x \leq 0)\&(y \leq 0) \\ (h_x - (x * f_W), h_y + (x * f_H)) & \text{if } (x \leq 0)\&(y > 0) \\ (h_x + (x * f_W), h_y + (x * f_H)) & \text{if } (x > 0)\&(y \leq 0) \\ (h_x + (x * f_W), h_y - (x * f_H)) & \text{if } (x > 0)\&(y > 0) \end{cases}$$

In these 2D frames, lines can be drawn to represent the trace of the players over the last *N frames* and the ball can also be animated using a basic linear motion model based on its tags. Moreover, in order to better understand the game by looking at the 2D representation (and once again, using the tags), the width of the squares corresponding to the players on court can be set to a larger value, and a cross can be drawn over those positions where a shot has been attempted.

D. Three-Player Selection

Although there are thousands of different basketball plays, there is a common pattern in some of them: even if the 5 players on court move during the action, only the movement of 3 of them is relevant for its outcome. For this reason, while extracting features of different plays, only data from 3 players is actually processed. It can be argued that typical plays like the *UCLA cut* or *Flex* are evident exceptions, but at the same time, there have been prestigious coaches (*e.g* Phil Jackson) who based their strategies in the *Triangle Offense*, a tactic first introduced by Sam Barry based on *three-player* concepts [18]. As explained in Section III-B, the players on court are known by reading event tags, and these have to be filtered in case there are more than 3 doing an exercise/playing the game. A simple assignation is done by matching the ID's of the involved players to the following variables: *Player1, Player2* and *Player3*.

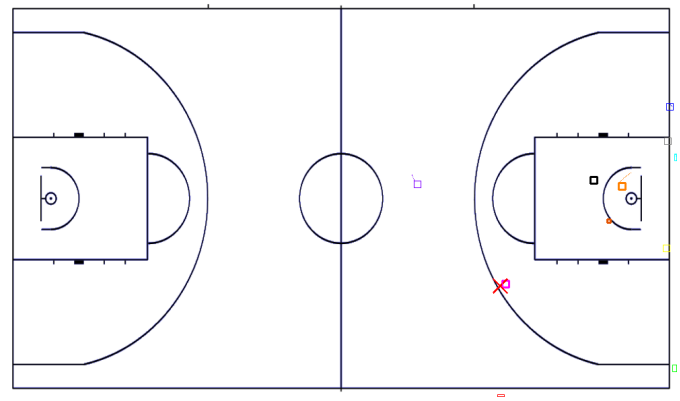

Fig. 3. 2D Representation of a *floppy offense* situation, executed by three players (pink, black and orange). As it can be seen, the player that corresponds to the pink square attempted a long-range shot and the ball (small orange square) is about the touch the rim.

Furthermore, players also have to be sorted by *positions*, which can be estimated with the players' coordinates at the beginning of the play; otherwise, patterns cannot be found in data, *e.g* small-fast players' features must be compared to other small-fast players and not to heavy-slow ones. The procedure is then to input a *region-map* that depends on the kind of situation and indicates where in the image and in which order the computer has to find those 3 relevant players. Some examples of *region-maps* are shown in Figure 4, where the orange, red and purple regions correspond to the zones where *Player 1, Player 2* and *Player 3* should be found respectively.

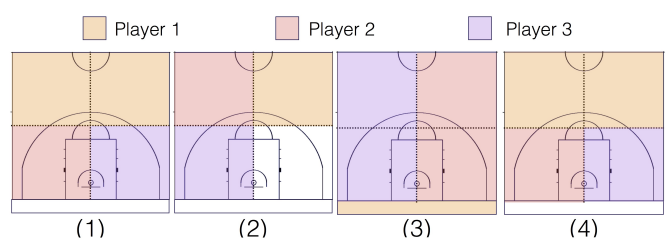

Fig. 4. Region Maps for the different types of exercises: (1) *Floppy Offense*, (2) Pick and Roll, (3) Press Break and (4) Post Up.

E. Feature Extraction

As the experimental dataset was limited in the number of observations, deep learning models such as Convolutional Neural Networks could not be applied. Therefore, features had to be manually extracted and carefully selected by an expert considering the game factors that allow distinguishing between two different plays. The basis of the feature extraction process is to have a single feature vector for each play containing both spatial and temporal information of the players. Note that this vector does not include any of the manually introduced ball information. Additionally, actions are divided into two segments in order to extract independent features from both the first and second half of the play, which usually contain non-correlated information. An example could be a *pick and*

roll sequence of duration T, where *Player1* calls the play at $t = 0$, receives the screen (see Appendix A) at $t = \frac{T}{2}$ and then drives to the basket. *Player1* was almost static in the first segment of the action $[0, \frac{T}{2}]$ but moved fast in the second one $[\frac{T}{2}, T]$; dividing plays into segments can help detecting these kind of behaviours.

For an action of duration T, each feature vector has a total of 51 features, including:

- The distance in meters between the basket and each player when $t = 0, t = \frac{T}{2}$ and $t = T$. The initial position is chosen because the player who calls the play must be sure that everybody is on their correct positions before executing it.
- The angle in degrees between the baseline and the line that goes from the basket to each player when $t = 0, t = \frac{T}{2}$ and $t = T$. Besides, the absolute angle is also computed in the same temporal conditions by calculating the angle between a parallel line to the sideline placed in the centre of the court and the line that goes from the basket to each player. The reason for including both types of angles is that many plays can be executed on both sides of the court, so the absolute angle adds robustness in this case.
- The total displacement of each player in both $[0, \frac{T}{2}]$ and $[\frac{T}{2}, T]$ segments, which indicates if the player is standing still or moving fast.
- The speed (in m/s) of every player in both $[0, \frac{T}{2}]$ and $[\frac{T}{2}, T]$ segments. This feature introduces temporal information to the vector and contextualizes the total displacement: if the displacement is high but the feature vector corresponds to a long play, speed shows that the movement is long but slow.
- The maximum distance in meters with respect to the basket of each player in both $[0, \frac{T}{2}]$ and $[\frac{T}{2}, T]$ segments. This feature is thought for detecting patterns in big players, which are used to play close to the basket; if the maximum distance of these kind of players is large, it is probably due to a screen they have set.
- The minimum distance between each pair of players in both $[0, \frac{T}{2}]$ and $[\frac{T}{2}, T]$ segments. Once again, this feature can help to identify if screens have been set during the play; moreover, it can also indicate the pair of involved players.

A visual representation of some features can be found in Figure 5.

IV. Results

A 96×51 matrix is obtained by extracting a 51-dimensional feature vector from each of the 96 observations. Nevertheless, considering the limited size of the dataset, the model cannot be trained as it is, because it might have non-relevant features that should not be learned, as it would lead to over-fitting. Principal Component Analysis (PCA) [7] is applied in order to reduce data dimensionality and discard those components that are highly correlated. The eleventh first principal components are kept in order to account for 95% of the variance in the data. This procedure is visually explained in Figure 6.

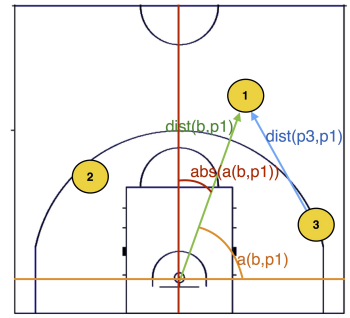

Fig. 5. Visual explanation of four features, being *dist(b,p1)* the distance from the basket to *Player1*, *dist(p1,p3)* the distance from *Player3* to *Player1*, and *a(b,p1)* and *abs(a(b,p1))* the angle and absolute angle respectively between the basket and *Player1*.

A 10-fold cross validation was used obtaining **97.9 % accuracy** when using a Linear Support Vector Machine classifier [6] with a *One-vs-One* strategy to deal with multiclass classification.

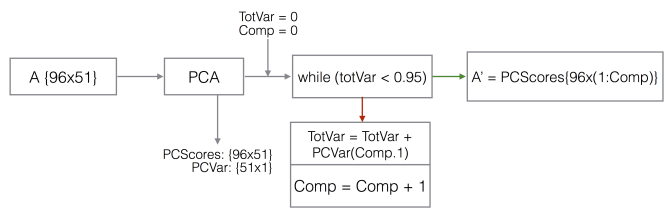

Fig. 6. Steps that have to be followed in order to retain 95% of the dataset's variance.

The resulting Confusion Matrix can be seen in Table I and a Scatter Plot can be found in Figure 7. Besides, Table II compares the obtained accuracy after using several Machine Learning algorithms over the dataset.

TABLE I
Confusion Matrix. Being: E1 - floppy offense, E2 - pick and roll, E3 - press break, E4 - post-up and E5 - fast break.

	E1	E2	E3	E4	E5
E1	**22**	0	0	0	0
E2	2	**20**	0	0	0
E3	0	0	**14**	0	0
E4	0	0	0	**21**	0
E5	0	0	0	0	**17**

V. Discussion

From the obtained results, it can be observed that the most conflicting situation is *press break*, and the reasoning behind is simple: while the objective of the other exercises is to repeat some movements by following certain patterns, the goal of this situation is simply to overcome pressure, no matter how; this fact makes the drill somewhat unpredictable.

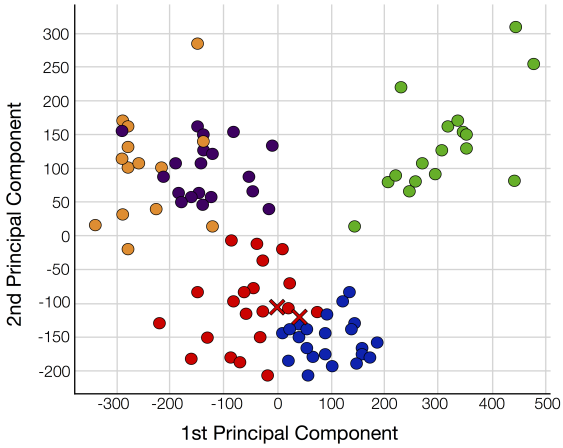

Fig. 7. Scatter Plot in 2D showing the well-classified (spots) and miss-classified (crosses) samples.

TABLE II
OBTAINED ACCURACY AFTER USING DIFFERENT MACHINE LEARNING ALGORITHMS OVER THE DATASET.

Algorithm	Accuracy (5-fold CV)	Accuracy (10-fold CV)
Simple Tree	83.3 %	83.3 %
Complex Tree	87.5 %	92.7 %
Linear Discriminant	**97.9** %	**97.9** %
SVM (linear)	**97.9** %	**97.9** %
SVM (cubic)	95.8 %	**97.9** %
Fine KNN	95.8 %	95.8 %
Bagged Trees	92.7 %	93.8 %

Despite the good performance of the classification model, a higher number of observations and classes would be required in order to build a professional system. Moreover, it would also be interesting to have subclasses, since the offense may change their strategy in real-time depending on their opponents defense and plays usually have second and third options (*i.e. pick and roll/pop*).

Once plays are correctly classified, advanced statistics can be extracted with ease: for example, the coach is able to know the pass speed of a certain player in all the *post-up situations* during a game.

Besides, manual ball tags proved to be useful, and not only to have the ball in a 2D representation, but also to temporally segment repetitions during the exercises. It might be argued that, while the whole purpose of the project is to substitute cameras for sensors, a camera has been used in the presented experiment; although it is a valid reasoning, the purpose of having a single camera is just to support sensor data and not to perform automatic tracking. Moreover, in the case of big companies, their camera setup includes a minimum of 6 high-quality fibre-synchronized cameras and, in the presented test, a simple camera was used (even a mobile phone recording could have been helpful). The best solution is adding a positioning sensor to the ball too, which must not change its weight. Actually, there are companies such as Wilson that are starting to commercialize this type of basketballs [17], so it is a feasible solution.

VI. CONCLUSIONS

In this article, a new method to automatically extract advanced statistics based on sensors data has been detailed. Even though working with sensors might have drawbacks (such as difficulties when trying to scout another team), it is a much cheaper solution than multi-camera configuration systems like the ones installed in NBA arenas, and it is a attainable way to start extracting advanced statistics in Europe. For the purposes of this publication, a dataset containing both video and tracking data of 30 minutes of the *Under-21 Valencia Basket Club*'s practice was recorded using NBN23's technological resources. In this recordings, there were a total of 96 different actions of the following classes (types of basketball plays): *floppy offense*, pick and roll, press break, post-up situations and fast breaks.

Knowing that the basis of the automatic extraction of statistics is the identification of different basketball plays occurring on court, these steps must be followed:

1) Labelling the frames containing events related to the ball using simple tags (receive, release, substitutions...).
2) Merging all tracking information in a single matrix, taking synchronization into account by sorting timestamp values.
3) For visualization purposes, mapping the players' court coordinates into pixels.
4) (For each play) Selecting three involved players in the action by introducing as an input a *region-map*.
5) (For each play) Extracting meaningful basketball features of the involved players in order to build a 1×51 feature vector.

In the presented test, once all feature vectors have been merged into the same data matrix, PCA has been applied in order to avoid model over-fitting, keeping the 95% of the observations' variance. Using a 10-fold cross validation and a Linear Support Vector Machine algorithm, 97.9% accuracy is obtained when trying to classify the whole training data.

A. Future Work

In order to improve the presented work, more data has to be recorded, containing a larger variety of observations and classes. Besides, it would be interesting to track the ball with a sensor instead of manual annotations. Likewise, more sequences corresponding to 5-on-5 games must be tested, as those actions will be less predictable; in addition, it would also be desirable to include defensive strategies. Another weakness of this project is the *region-map* input used to find the 3-related players; this map should generalize to any kind of sequence. An alternative could be using techniques such as dynamic time warping to align trajectories and find out correlated signals . With thousands of examples, Convolutional Neural Networks would provide higher accuracy and data could be divided into training and testing

sets.

Another interesting purpose could be applying the same technique to recognize patterns in other sports, especially in soccer, where European clubs have high salary caps.

Appendix A
Basketball Glossary

This appendix is devoted to provide a detailed description of some technical basketball concepts that are relevant to further understand some of the assumptions of the study.

A. Screens

Screens are usually set by big-players, which stay static in a certain position in order to retain the defender of a guard (fast-small players). Therefore, the small player can take advantage of the lack of a defender for few seconds.

B. Plays

In order to explain the plays that have been included in the gathered dataset, pictorial representations (like the ones coaches draw in their boards) are shown. Using the icons shown in Figure A8, three temporal frames are shown for each play, which explain the movement that is going on during the action. The representations of *floppy offense*, pick and roll, press break and post-up situations can be seen in Figures A9, A10, A11, A12 and A13 respectively.

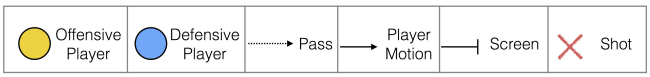

Fig. A8. Icons used in the 2D representations of basketball plays.

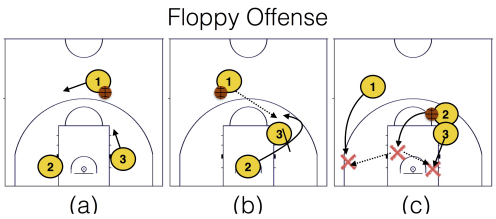

Fig. A9. Temporal execution of *floppy offense*; (a) *Player 1* creates space and *Player 3* gets prepared to set a screen; (b) *Player 3* screens away *Player 2*, who receives the ball and drives to the basket; (c) *Player 2* ends up deciding if he/she shoots, looks for an open shot of *Player 1* or the roll of *Player 3*.

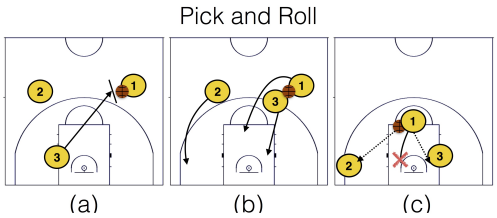

Fig. A10. Temporal execution of a pick and roll sequence; (a) *Player 1* calls the play and *Player 3* sets him a screen; (b) *Player 1* drives to the basket, *Player 2* looks for a comfortable spot in the corner and *Player 3* continues to the basket; (c) *Player 1* ends up deciding if he/she shoots, looks for an open shot of *Player 2* or the roll of *Player 3*.

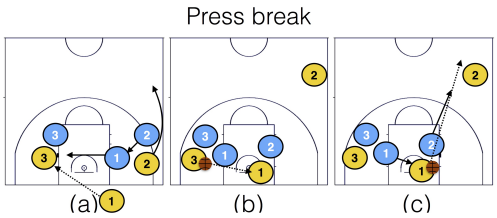

Fig. A11. Temporal example of a press break situation, where the pass of *Player 1* to *Player 2* overcomes the defensive pressure. Note that there is not a universal way of breaking pressure, as it depends on the defensive team's reaction.

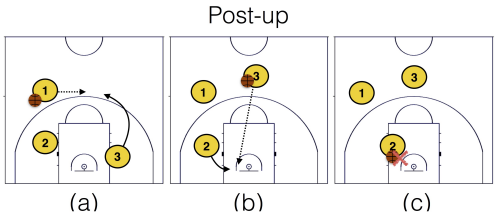

Fig. A12. Temporal execution of a post-up situation; (a) *Player 1* tries to pass the ball to *Player 2*, but he/she is being guarded, so *Player 3* looks for a better passing position; (b) with a better angle, *Player 2* gives an assist to *Player 3*, who (c) ends up shooting from a close position to the basket.

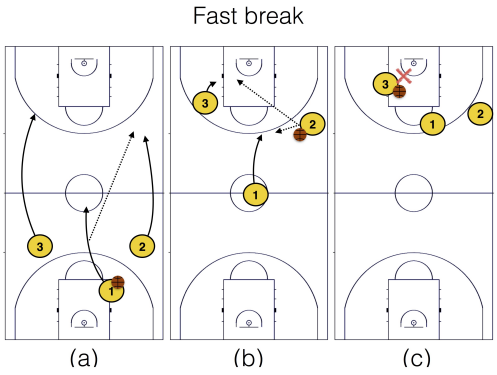

Fig. A13. Possible temporal execution of a fast break; (a) *Player 1* tries steals the ball, while *Player 2* and *Player 3* run to the offensive end; (b) *Player 2* receives in the three point line, and can (c) pass to *Player 3*, who arrived first, or wait for the cut of *Player 1*.

REFERENCES

[1] J.-Y. Bouguet. Pyramidal implementation of the affine lucas kanade feature tracker description of the algorithm. *Intel Corporation*, 5(1-10):4, 2001.

[2] F. Er, B. Dežman, G. Vu, J. Perš, M. Perše, and M. Kristan. An Analysis of Basketball Players Movements in the Slovenian Basketball League Play-Offs Using the SAGIT Tracking System. *Physical Education and Sport*, 6(1):75–84, 2008.

[3] ESPN. NBA extends television deals with ESPN, TNT. http://www.espn.com/nba/story/_/id/11652297/nba-extends-television-deals-espn-tnt; Last Accessed: 2017-04-29.

[4] P. Felsen and P. Lucey. Body Shots : Analyzing Shooting Styles in the NBA using Body Pose. In *MIT Sloan, Sports Analytics Conference*, 2017.

[5] HoopsHype. NBA Salaries. http://hoopshype.com/salaries/; Last Accessed: 2017-04-29.

[6] C.-W. Hsu and C.-J. Lin. A comparison of methods for multiclass support vector machines. *IEEE transactions on Neural Networks*, 13(2):415–425, 2002.

[7] I. Jolliffe. *Principal component analysis*. Wiley Online Library, 2002.

[8] P. Lucey, A. Bialkowski, P. Carr, Y. Yue, and I. Matthews. How to Get an Open Shot : Analyzing Team Movement in Basketball using Tracking Data. In *MIT Sloan, Sports Analytics Conference*, 2014.

[9] A. C. Miller and L. Bornn. Possession Sketches : Mapping NBA Strategies. In *MIT Sloan, Sports Analytics Conference*, 2017.

[10] NBN23. Technology to take Basketball into the future. https://www.nbn23.com/en/; Last Accessed: 2017-04-29.

[11] M. Perše, M. Kristan, J. Perš, and S. Kovacic. A Template-Based Multi-Player Action Recognition of the Basketball Game. In *CVBASE '06 - Proceedings of ECCV Workshop on Computer Vision*, pages 71–82, 2006.

[12] V. Ramanathan, J. Huang, S. Abu-el haija, A. Gorban, K. Murphy, and L. Fei-fei. Detecting events and key actors in multi-person videos. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

[13] Second Spectrum. Our Work. https://www.secondspectrum.com/ourwork.html; Last Accessed: 2017-04-29.

[14] STATS. Basketball Data Feed — Basketball Player Tracking — SportVU.

[15] C. Stauffer and W. E. L. Grimson. Adaptive background mixture models for real-time tracking. In *Computer Vision and Pattern Recognition, 1999. IEEE Computer Society Conference on.*, volume 2, pages 246–252. IEEE, 1999.

[16] K.-c. Wang and R. Zemel. Classifying NBA Offensive Plays Using Neural Networks. In *MIT Sloan, Sports Analytics Conference*, pages 1–9, 2016.

[17] Wilson. Wilson X Connected Basketball - Smart Sensor Basketball & App. http://www.wilson.com/en-us/explore/basketball/wx; Last Accessed: 2017-04-29.

[18] T. Winter. The Encyclopedia of the Triangle Offense, 2006.