Proceedings of the Sixth Australian Conference on MATHEMATICS AND COMPUTERS IN SPORT

Bond University, Queensland

Edited by Graeme Cohen and Tim Langtry
Department of Mathematical Sciences
Faculty of Science
University of Technology, Sydney

6M&CS

1 – 3 July 2002

Contents

Conference Director's Report v

Participants vi

Program vii

Principal Speakers

Cricketing chances
G. L. Cohen 1

How to fix a one-day international cricket match
Stephen Gray and Tuan Anh Le 14

The fundamental nature of differential male/female world and Olympic winning performances, sports and rating systems
Ray Stefani 32

Contributed Papers

Factors affecting outcomes in test match cricket
Paul Allsopp and Stephen R. Clarke 48

Predicting the Brownlow medal winner
Michael Bailey and Stephen R. Clarke 56

Using Microsoft® Excel to model a tennis match
Tristan J. Barnett and Stephen R. Clarke 63

Studying the bankroll in sports gambling
D. Beaudoin, R. Insley and T. B. Swartz 69

How do lawn bowls and golf balls slow down on grass?
Maurice N. Brearley and Neville J. de Mestre 78

Fixing the fixtures with genetic algorithms
George Christos and Jamie Simpson 92

Collecting statistics at the Australian Open tennis championship
Stephen R. Clarke and Pam Norton 105

Dynamic evaluation of conditional probabilities of winning a tennis match
Shaun Clowes, Graeme Cohen and Ljiljana Tomljanovic 112

Analysing scores in English premier league soccer
John S. Croucher 119

Review of the application of the Duckworth/Lewis method of target resetting in one-day cricket
Frank Duckworth and Tony Lewis 127

Can the probability of winning a one-day cricket match be maintained across a stoppage?
Frank Duckworth and Tony Lewis 141

A biomechanical power model for world-class 400 metre running
Chris Harman 155

Analysis of a twisting dive
K. L. Hogarth, M. R. Yeadon and D. M. Stump 167

Tennis serving strategies
Graham McMahon and Neville de Mestre 177

Team ratings and home advantage in SANZAR rugby union, 2000–2001
R. Hugh Morton 182

Soccer: from match taping to analysis
Emil Muresan, Jelle Geerits and Bart De Moor 188

Video analysis in sports: VideoCoach©
Emil Muresan, Jelle Geerits and Bart De Moor 196

Serving up some grand slam tennis statistics
Pam Norton and Stephen R. Clarke 202

Optimisation tools for round-robin and partial round-robin sporting fixtures
David Panton, Kylie Bryant and Jan Schreuder 210

The characteristics of some new scoring systems in tennis
Graham Pollard and Ken Noble 221

The effect of a variation to the assumption that the probability of winning a point in tennis is constant
Graham Pollard 227

A solution to the unfairness of the tiebreak game when used in tennis doubles
Graham Pollard and Ken Noble 231

Dartboard arrangements with a concave penalty function
E. Tonkes 236

Abstract

Diving into mathematics or some mathematics of scuba diving
Michel de Lara 245

SOCCER: FROM MATCH TAPING TO ANALYSIS

Emil Muresan, Jelle Geerits and Bart De Moor
ESAT-SCD, KULeuven
Kasteelpark Arenberg 10
3001 Heverlee (Leuven)
Belgium
emil-mihai.muresan@esat.kuleuven.ac.be, jelle.geerits@esat.kuleuven.ac.be,
bart.demoor@esat.kuleuven.ac.be

Abstract

Soccer games analysis has so far been done only by watching the recording of the game and making remarks about the interesting elements in it. There is no exact data to use, let alone detailed and quantitative analysis of player and team performances and tactics.

This paper proposes a different approach to match analysis, using cameras and computers to extract player coordinates throughout the whole match. Once we have the coordinates database, any statistics can be inferred, e.g. on individual player performance (position, speed, acceleration, field coverage, etc.) as well as on team tactics (players' complementarity, cohesion, field coverage, etc.).

1 Overview

In this paper we propose a full technological process for match analysis. As it is a proposal, not all the steps are implemented and there are still problems to be tackled. But the potential of such an analysis method is huge, as will be pointed out.

The process by which we obtain the desired analysis and statistics of a soccer match has several steps: match taping, digitising the video, coordinate extraction, and full analysis on the coordinates database. Each of these steps will be described in the following. The whole process is depicted in Figure 1.

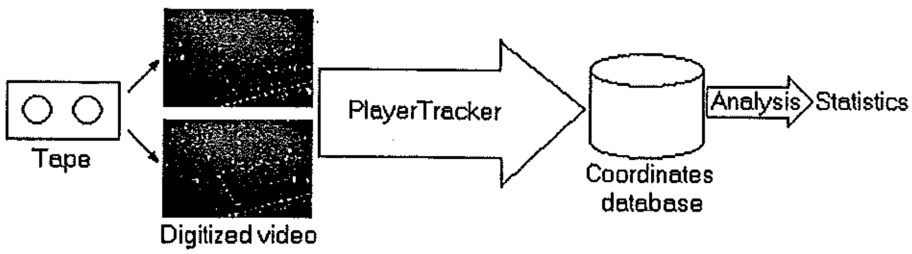


Figure 1: The whole analysis process.

2 Match taping and digitisation

We start from the video recording of a soccer match on tape. Taping of the match is done with four fixed cameras from the four corners of the field (Figure 2), each surveying a quarter (in order to get sufficient video quality for the next step). The cameras are fixed, static ones. This is required in order to calculate the mapping between the position of the player in the image and its real position on the field.

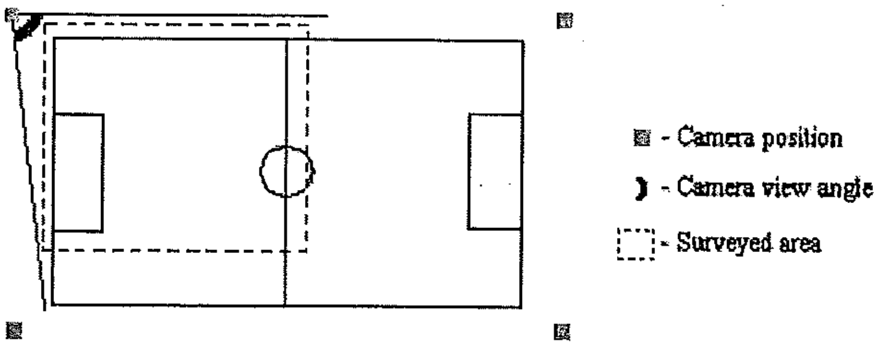


Figure 2: Camera positions for match taping.

After the whole match is recorded on tape, it has to be digitised and stored as a video file (avi). Once we have the digital recording of the match, the coordinate extraction tool can be applied.

The digitised videos can have any resolution as far as the extraction tool is concerned, but there are limitations due to the minimum image quality needed for being able to identify the players in the images. For digitisation we used Dazzle MovieStar (www.dazzle.com), a very easy-to-use multimedia tool that uses some dedicated hardware to do digitisation and video compression in real time. Some parameters that we used in the digitisation process are: MPEG-2 video encoder, image size not bigger than 720×480 pixels, a sampling rate of 25 frames per second, and colour depth of 24 bits per pixel.

The processing of these images does not run in real time. The application is run on a computer with a 500MHz processor and 512MB of RAM, using an image processing algorithm that achieves a compromise between accuracy and speed, resulting in a processing rate of ten frames per second, that is, 2.5 times slower than real time.

3 Coordinate extraction

PlayerTracker is a tool that we developed especially for this step of the process, i.e. coordinates extraction. It uses the images digitised in the previous step. Those images are processed and the players are automatically tracked throughout the match. The coordinates of the players are saved in a database for exploitation in the next step (analysis).

The PlayerTracker's objective is to obtain the coordinates of the players, the referee and the ball during the match. This tool makes use of image processing techniques to track each player during the whole match and compute their positions on the field in every frame of the video recording.

The coordinate extraction process is a cycle. A frame is taken from the video, players are detected in this frame, and the real coordinates of these players on the real field are computed and saved in the database. After this we move on to the next frame.

To make the detection easier and faster, player position information from the previous frame is used to detect the players in the current frame. We make use of two rectangles to track each player: one rectangle has the dimensions of the player and is centred on the player, the second bigger rectangle is given such dimensions that the player cannot exit that rectangle in between two consecutive frames (because of physical limitations).

In the beginning, some user interaction is needed for initialisation. The user must specify the mapping between the field in the perspective view of the video and the real field. This mapping will be used to translate the detected player position from image coordinates to real physical coordinates on the field. Next, because information from a previous frame is not available, the user has to initialise the system by interactively placing (e.g. mouse click and drag in the image) the tracking rectangles on the players. Then, the detection is done automatically (apart from some exceptions that will be discussed later in this paper). The system after the initialisation phase is shown in Figure 3. The three players to be tracked (left image) are indicated by the tracking rectangles. Their physical positions on the field are computed and visualised in the "virtual" top view of the field (right image).

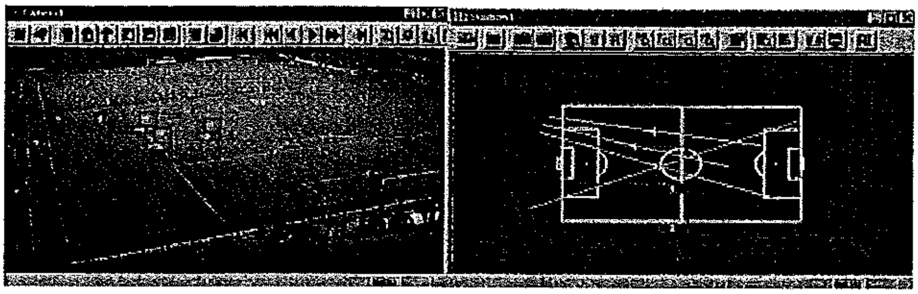


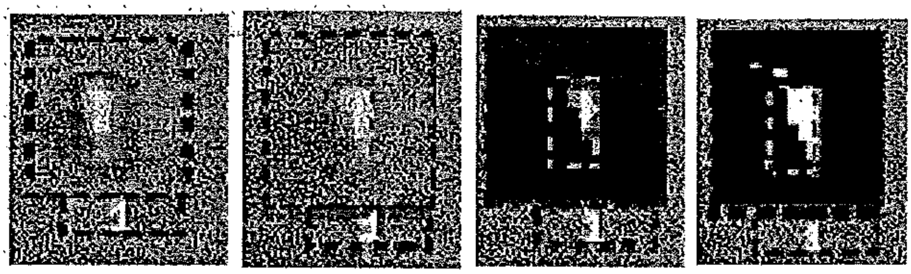
Figure 3: The detection system after initialisation.

Using the tracking rectangles makes the detection faster and more robust because each player is searched for only inside their own rectangle, and not in the whole frame image.

For each player the same detection process is applied. The image inside the bigger tracking rectangle is preprocessed (smoothing, contrast enhancement). Then image processing and object detection algorithms are used for the detection of the player.

The algorithms used to detect a player in a rectangle employ different image processing techniques for contrast enhancement, smoothing, edge detection (Sobel and Prewitt operators) and segmentation [2]. The actual detection algorithms are based on pattern matching, histogram analysis, dynamic thresholding, etc. Their performance differs and is dependent on the image quality. The images on which the tests were made are not of very high quality, because one of the objectives of our project is to keep the whole procedure at low cost, so the images were taken with ordinary cameras.

The detection can be performed either on gray-scale or on colour images. One of the algorithms for gray-scale images with its subsequent steps (a. initial image; b. noise elimination with a Max kernel and enlargement of the object; c. gray scale stretching, contrast enhancement; d. binarisation, object detection is straightforward) is exemplified in Figure 4.



a. b. c. d.

Figure 4: Player detection in gray-scale images (a. initial image; b. noise elimination with a Max kernel and enlargement of the object; c. gray scale stretching, contrast enhancement; d. binarisation).

An example of player detection algorithm for colour images is the detection by histogram comparison.

The small detection rectangle parses the bigger rectangle, and for each position its colour histogram is computed and it is compared with the reference histogram of the empty field. The position for which the histogram is the most distant from the reference is considered to be the player.

In Figure 5 an example of the histograms for the reference and player rectangles is given. Histograms are computed on the inner tracking rectangle, and displayed separately for each of the three colour components (R, G, B). Reference image histograms are more peak-like because they are quite uniform (green); player histograms are less uniform due to equipment color.

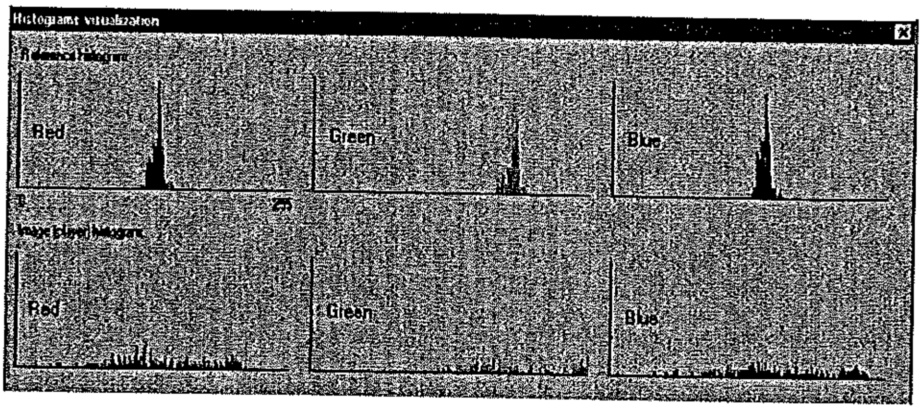


Figure 5: Histogram comparison between the reference image (first row) and the player image (second row).

Several metrics to compute the "distance" between two colour histograms were taken into consideration. One is the Kullback–Leibner distance [1] that is widely used in probability distribution problems. This is

$$D(p,q) = \sum_{n \in X} p(n) * \log \frac{p(n)}{q(n)},$$

where p and q are two discrete distributions over the domain X.

In this case, the two distributions are the concatenated colour histograms (R,G,B) of the reference (grass) and the player image at the current position of the detection rectangle.

However, because of the computational complexity of this metric, a simpler one was chosen:

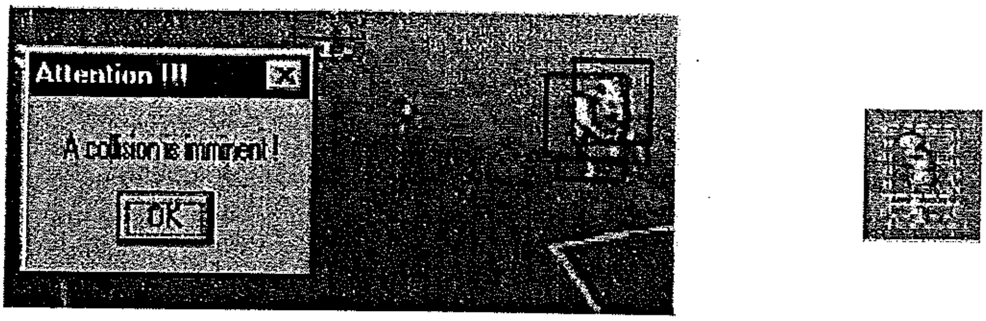
$$D(RH, CH) = \sum_{i \in [0,255]} (|RHr[i] - CHr[i] + |RHg[i] - CHg[i]| + |RHb[i] - CHb[i]|)$$

where RHr, RHg, RHb are the R, G, B histograms of the reference (grass) image, and CHr, CHg, CHb are the R, G, B histograms of the detection rectangle at the current position.

After having obtained the position of the player in the current frame in the video, the mapping to the real field coordinates is used to compute the player's real position on the field. These real field coordinates are then saved in the database.

The mapping function maps the perspective polygonal view of the scaled field onto the real rectangular field. The user is required to graphically make the correspondence between the field lines in the image and field projection during the initialisation phase.

Of course there are also some problems in the process and not everything goes smoothly from one whistle to another. Player collisions make this task more difficult than it seems. Some algorithms were developed to deal with several of the collision situations, but for some cases user interaction is still needed. Figure 6 shows an example of how the software deals with the collision problem. The application detects when a collision is about to occur (tracked players are lost) and displays a warning (Figure 6a). If a collision has occurred (Figure 6b) the detection is stopped with the tracking rectangles in the last known positions and the user has to reinitialise the colliding players.



a. b.

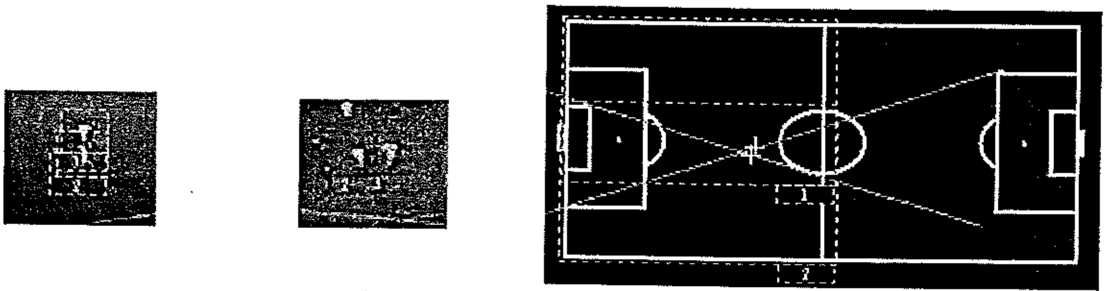
Figure 6: Player collision solving.

The algorithms for player collision solving are based on previous player positions, multiple cameras information, player equipment colour. But if the colliding players overlap completely in the image, the detection process cannot discriminate between them.

Collisions are detected from the intersections of the tracking rectangles. If the inner rectangles of two players intersect more than a certain percentage (a threshold), a collision is signalled and the corresponding solving algorithms are enabled.

One type of collision solving is to use two cameras surveying the same portion of the field. While for one camera two players appear to collide because they are on the same line with the camera, the other camera from a different angle can clearly differentiate between them.

One such situation is shown in Figure 7, as well as the way in which two cameras with different orientations can solve this problem. Two players are tracked in this examples (each player's number is displayed at the bottom of the tracking rectangle; in Figure 7 the players with numbers 1 and 2 are tracked). Figure 7a shows the two players from the viewpoint of the first camera (the two players are colliding). Viewing the same scene from the viewpoint of the second camera (Figure 7b), the two players can clearly be discriminated. Figure 7c shows the real field positions of the two players, computed using information from both cameras (here numbers in the figure specify the two cameras surveying the corresponding regions; the two crosses show the player positions).



a. b. c.

Figure 7: Collision solving with two cameras; a. first camera; b. second camera.

Even if players are not colliding, the two cameras are useful for a more precise computation of the real player position on the field (player positions on the field are displayed in Figure 7c). Normally, there are small errors in the mapping from image to real field, and these can be corrected a certain amount by averaging in between the coordinates computed from the two different cameras (sensor fusion).

4 Analysis based on the coordinates database

Once the coordinates are extracted, quantitative game analysis can be performed (different analysing methods that can be priceless to soccer coaches), and a virtual reconstruction of the match can be made for analysis, entertainment or commercial purposes.

All kinds of statistics can be calculated such as: field coverage, total distance run by a player, velocity, acceleration, ball possession percentage, ball losses, etc.

Even tactics can be built in. There is, for instance, a claim that when you take the polygon that wraps the defensive players (the "convex hull") and calculate its centre of gravity, and you do the same thing for the offensive players of the other team, you can determine, by the relative position of the body mass point if there is danger for a counter-attack [3]. The coach can then be alerted about this situation and give the team directions to be careful in the future. The player polygon can easily be drawn from the player coordinates at each moment of the game, and displayed as in Figure 8.

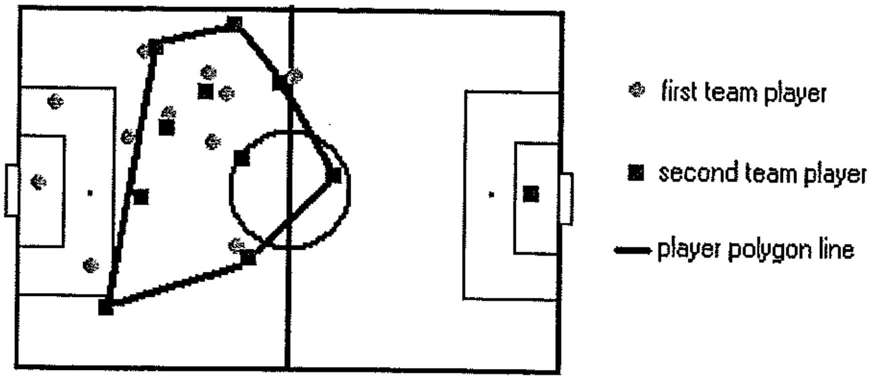


Figure 8: The player polygon ("convex hull") is the smallest polygon that encloses all players of a team.

Another possible use of the coordinates is 3D replay. A situation can be replayed and the user can have a look at the scenario from all angles (e.g. how did the goal-keeper see the ball).

A first scheme of what kind of analysis can be performed once the coordinates are extracted is given below to emphasise the importance of this process. This is a kind of knowledge tree with respect to possible kinds of analysis:

Analysis scope:
- per player
- per group of players
  - attack
  - midfield
  - defence
- per team
- per club

Time horizon:
- per phase
- per *a priori* defined game period
- per game
- over different games

Statistics:
- ball possession per team
- time/position tensor
- tracking the line-up through a match (4:4:2 or 3:2:5 or 5:2:3)

Markov model:
- gives the *a posteriori* statistics of passes
- by deriving the Markov model of both teams:
  - ball possession, loss and recuperation can be quantified
  - per group of players a passing matrix can be made to show their complementarity

Goal keeper:
- visualise the goal attempts on his goal
- distant or close shots / good or bad keeper reactions
- preferential directions

Per individual player:
- physical condition
- run distance
- speed profiles
  - double acceleration
  - explosivity
- run lines
- position per unit of time
- strategy and tactical insight of the game:
  - goes deep in the field
  - works hard
  - can anticipate well (e.g. positioning with respect to the others)

For the referee:
- track/monitor the referee performance
- position on the field in relation to field situations

This is only a coarse categorisation of possible analysis methods, and the elements mentioned above will be detailed and completed.

5 Future work

In the future there will be two distinct approaches for the tracking. The use of fixed cameras has proven to be useful and will be improved, but also moving cameras will enter the scene. The use of the moving cameras is already tested in reality for ball tracking purposes. Here the pan and tilt of the camera is measured using the optical mouse principle.

Another use for the moving camera is to determine the exact identity of a player after a collision that may have caused a switch of identity. Therefore, the moving camera scans the field and if a player whose identity is uncertain is found, the camera follows him until the back number can be read.

Acknowledgments

Dr Bart De Moor is a full professor at the Katholieke Universiteit Leuven, Belgium. Our research is supported by grants from several funding agencies and sources: Research Council KUL: Concerted Research Action GOA–Mefisto 666 (Mathematical Engineering), IDO (IOTA Oncology, Genetic networks), several

PhD/postdoctoral and fellowship grants; Flemish Government: Fund for Scientific Research Flanders (several PhD/postdoctoral grants, projects G.0256.97 (subspace), G.0115.01 (bio-i and microarrays), G.0240.99 (multilinear algebra), G.0197.02 (power islands), G.0407.02 (support vector machines), research communities ICCoS, ANMMM), AWI (Bil. Int. Collaboration Hungary/Poland), IWT (Soft4s (softsensors), STWW–Genprom (gene promotor prediction), GBOU–McKnow (Knowledge management algorithms), Eureka–Impact (MPC–control), Eureka–FLiTE (flutter modeling), several PhD grants); Belgian Federal Government: DWTC (IUAP IV-02 (1996–2001) and IUAP V-10-29 (2002–2006): Dynamical Systems and Control: Computation, Identification & Modelling), Program Sustainable Development PODO-II (CP-TR-18: Sustainability effects of Traffic Management Systems); Direct contract research: Verhaert, Electrabel, Elia, Data4s, IPCOS.

References

[1] S. Kullback, *Information Theory*, Dover, New York (1968).

[2] V. S. Nalwa, *A Guided Tour of Computer Vision*, Addison–Wesley (1993).

[3] F. Masson, *Football +: Comment gerer une equipe?* manual from the Belgian Heizel Trainer School, Brussels (1998).