Non-Invasive Soccer Goal Line Technology: A Real Case Study

Paolo Spagnolo, Marco Leo, Pier Luigi Mazzeo, Massimiliano Nitti, Ettore Stella, Arcangelo Distante
National Research Council of Italy
spagnolo@ba.issia.cnr.it

Abstract

In this paper, a real case study on a Goal Line Monitoring system is presented. The core of the paper is a refined ball detection algorithm that analyzes candidate ball regions to detect the ball. A decision making approach, by means of camera calibration, decides about the goal event occurrence. Differently from other similar approaches, the proposed one provides, as unquestionable proof, the image sequence that records the goal event under consideration. Moreover, it is non-invasive: it does not require any change in the typical football devices (ball, goal posts, and so on). Extensive experiments were performed on both real matches acquired during the Italian Serie A championship, and specific evaluation tests by means of an artificial impact wall and a shooting machine for shot simulation. The encouraging experimental results confirmed that the system could help humans in ambiguous goal line event detection.

1. Introduction

Soccer is the world's most popular sport and an enormous business, and every match is currently refereed by a single person who "has full authority to enforce the Laws of the Game". So, controversies are inevitable, and the most glaring of them are usually about referee calls for which no interpretation is required and concern about whether the ball has completely crossed goal line or not. Recently, famous 'bad calls' happened during the Euro 2012 (Ukraine scored a goal against England that clearly went over the line but was disallowed by referee, see fig. 1) and World Cup 2010 (England scored a goal against Germany that was disallowed by referee) Competitions.

In cases like these, the referee's call is influenced by, among other things, three ineluctable factors:

- the referee's position on the field: he is not aligned with the goal line and then a parallax error affects his decision;
- the high speed of the ball that can reach up to 120km/h. It is impossible for human visual and cognitive systems (as well as for standard broadcast images, at 25fps) to estimate the position of such a moving object continuously.
- the considerable distance (about 35-40 m.) between the linemen and the goal post: this makes it very hard to evaluate goal events with a resolution of about 1-2 cm.

Figure 1. During the 2012 Euro Competition, England's defender John Terry lunges for the ball, which appears to be over the line

The only way to definitively avoid these kinds of controversies is to introduce a "goal line technology", i.e an automatic system to assist the referee in decisions concerning goal events.

For this purpose different technologies have been proposed. The earliest ones were based on instant replay: in case of a controversial call about a goal event the referee (or an assistant) could stop the game and watch the images (acquired from broadcast or dedicated cameras). This would slow down the game taking away possible plays and annoying the audience. Thus attention has recently turned to technologies able to decide autonomously whether or not the ball has crossed the goal line. One of the most promising approaches uses a magnetic field to track a ball with a sensor suspended inside [3]. Thin cables with electrical current running through them are buried in the penalty box and behind the goal line to make a grid. The sensor in the ball measures the magnetic grids and relays the data to a computer which determines if the ball has crossed the line or not. However, this kind of technology cannot provide unquestionable proof of detected events; and requires substantial modifications to the infrastructure of the stadium and game component (ball, playing-field, goalposts,...).

For these reasons, the efforts of several companies and research institutes are currently focused on the development of non-invasive goal line technologies. In particular, vision-based systems appear to be very promising considering their capability to provide a posteriori verification of the system's operations [1, 2].

The main issue of an automatic system is the detection of the ball; it is very difficult when images are taken from fixed or broadcast cameras with a wide camera view since the ball is represented by a small number of pixels and moreover it can have different scales, textures and colors. For this reason, most ball detection approaches are based on an evaluation of the ball trajectory. The underlying idea is that the analysis of kinematic parameters can point out the ball among a set of ball candidates [13, 15, 11, 10].

However, trajectory based approaches are generally off-line since the evaluation of the kinematic parameters for all ball candidates requires a long period of observation; so they are not suitable to be used in a real time goal line monitoring system.

In recent years, few research groups have started working on visual frameworks with the aim of recognizing real time events. These systems have also to address problems associated with the time spent on image acquisition, transmission and processing (often the frame rate is even higher than for standard TV cameras). Furthermore, the ability to work autonomously for several hours and in all environmental conditions are additional characteristics required in this kind of systems. In [5] the authors present a real time visual system for goal detection which can be used as decision support by the referee committee. A system for automatic judgment of offside events is presented in [7]. The authors propose the use of 16 cameras located along both sides of the soccer field to cover the whole area. The integration of results from multiple cameras is used for offside judgment. Six fixed cameras were used in [4] to cover the whole field and to acquire image sequences from both sides of the stadium. Player and ball tracking processes run parallel on the six image sequences and extract the player and ball positions in real time.

However, the ball detection approaches proposed in these works are developed to perform mainly in single image; they don't use temporal consistency to reinforce the detection, and also integration between different views is quite superficial. In our work we integrate all information to realize a system able to work consistently for long time periods.

In this paper, a visual system able to detect the goal event through real time processing of the acquired images and immediately provide the image sequence that records the goal event under consideration is presented. The system has been implemented at the Friuli Stadium in Udine. It has been tested both during real matches of the Italian Serie A championship, and specific simulation sessions: in this case, the ball was shot by a shooting machine in different contexts, as explained in detail in the experimental results section, in order to validate the system in terms of both spatial and temporal accuracy.

2. Overview of the System

In figure 2(a) the visual system is outlined. Six cameras are placed on the stands of the stadium. For each side of the pitch, two of the three cameras have their optical axes parallel to the goal frame, the remaining one is placed behind the goal with its optical axis perpendicular to the goal frame. Each camera is connected to a processor (node) that records and analyzes the acquired images. In figure 2(b) a schematic diagram of the processing steps executed by each node is shown.

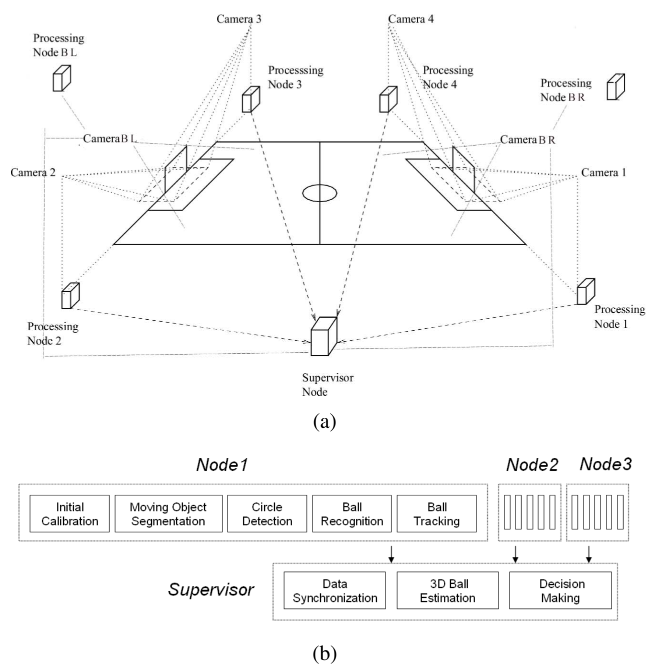


Figure 2. The scheme of the visual system (2(a)), and a schematic diagram of the processing steps executed by each Node (2(b))

The six processors are connected to a main node, which has the supervisor function. The supervisor node has a decision making function by combining the processing results coming from the cameras. The strategy is based on some heuristics that perform data fusion evaluating the time space coherence of the ball's 3D trajectory. The processing results of the three corresponding nodes are compared and a goal event probability function is evaluated.

3. Preliminary Steps: Calibration and Moving Object Segmentation

First of all, it is necessary to do a calibration step for each node in which the correspondences between the image plane and a plane in the 3D world are assessed. This step is fundamental in determining the 3D position of the ball. In other words, the homography transformation matrix is estimated by using Random Sample Consensus for each node [6] in this step. Each homography transformation matrix $\mathbf{M_i}$ relates the points on the image plane to the corresponding points on the 3D plane. The only constraint to be considered when choosing the planes is that they must not be perpendicular to the image plane of the associated camera. This calibration needs to be done only once, after camera installation, and if the cameras remain in place these measures are still valid for any subsequent matches. For the experiments reported in this paper, the calibration phase was carried out using non-deformable steel structures: each structure defines a plane in the 3D world and specific markers were used for the identification of the control points.

The segmentation of the image to extract moving objects is the first processing step executed by each node. It is fundamental as it limits the ball detection to moving areas and reduces computational time. For this purpose a background subtraction-based segmentation algorithm was implemented. Firstly, a background model has to be generated and then continuously updated to include lighting variations in the model. The implemented algorithm uses the mean and standard deviation to provide a statistical model of the background. Detection is then performed by comparing the pixel current intensity value with its statistical parameters, as explained in several works on this topic (a good review can be found in [12]). Details about the implemented approach can be found in [14] Finally, after the detection of moving points, a connected components analysis detects the blobs in the image by grouping neighboring pixels. After this step, regions with an area less than a given threshold are considered as noise and removed, whilst remaining regions are evaluated in the following steps.

4. Ball Detection and Tracking

An automatic method that detects ball position in each image is the central step to building the vision system. In the soccer world, a great number of problems have to be managed, including occlusions, shadowing, mis-detection (the incorrect detection of objects similar to the ball), and last but not least, real time processing constraints. The ball detection method has to be very simple, fast and effective as a great number of images per second must be processed. This kind of problem can be addressed by considering two different detection systems: geometric approaches that can be applied to match a model of the object of interest to different parts of the image in order to find the best fit; or example based techniques that can be applied to learn the salient features of a class of objects from sets of positive and negative examples.

This method uses two different techniques together in order to take advantage of their peculiarities: first of all, a fast circle detection (and/or circle portion detection) algorithm, based only on edge information, is applied to the whole image to limit the image area to the best candidate containing the ball; second, an appearance based distance measure is used to validate ball hypothesis.

The Circle Hough Transform (CHT) aims to find circular patterns of a given radius R within an image. Each edge point contributes a circle of radius R to an output accumulator space. The peak in the output accumulator space is detected where these contributed circles overlap at the center of the original circle. In order to reduce the computational burden and the number of false positives typical of the CHT, a number of modifications have been widely implemented in the last decade. The use of edge orientation information limits the possible positions of the center for each edge point. This way only an arc perpendicular to the edge orientation at a distance R from the edge point needs to be plotted. The CHT, as well as its modifications, can be formulated as convolutions applied to an edge magnitude image (after suitable edge detection). We have defined a circle detection operator that is applied over all the image pixels, which produces a maximal value when a circle is detected with a radius in the range [Rmin,Rmax]:

$$u(x,y) = \frac{\int\int_{D(x,y)} \vec{e}(\alpha,\beta)\cdot\vec{O}(\alpha-x,\beta-y)d\alpha d\beta}{2\pi(R_{max}-R_{mim})} \quad (1)$$

where the domain D(x,y) is defined as:

$$D(x,y) = \{(\alpha,\beta)\in\Re^2 | R_{min}^2 \leq (\alpha-x)^2+(\beta-y)^2 \leq R_{max}^2\} \quad (2)$$

$\vec{e}$ is the normalized gradient vector:

$$\vec{e}(x,y) = [\frac{E_x(x,y)}{|E|},\frac{E_y(x,y)}{|E|}]^T \quad (3)$$

and $\vec{O}$ is the kernel vector

$$\vec{O}(x,y) = [\frac{\cos(\tan^{-1}(y/x))}{\sqrt{x^2+y^2}},\frac{\sin(\tan^{-1}(y/x))}{\sqrt{x^2+y^2}}]^T \quad (4)$$

The use of the normalized gradient vector in (1) is necessary in order to have an operator whose results are independent from the intensity of the gradient in each point: we want to be sure that the circle detected in the image is the most complete in terms of contours and not the most contrasted in the image. Indeed, it is possible that a circle that is not well contrasted in the image gives a convolution result lower than another object that is not exactly circular but has a greater gradient. The kernel vector contains a normalization factor (the division by the distance of each point from the center of the kernel) which is fundamental to ensuring that we have the same values in the accumulation space when circles with different radii in the admissible range are found. Moreover, normalization ensures that the peak in the convolution result is obtained for the most complete circle and not for the greatest in the annulus. As a final consideration, in equation (1) the division by $(2\Pi \cdot (R_{max} - R_{min}))$ guarantees the final result of our operator in the range [-1,1] regardless of the radius value considered in the procedure. The masks implementing the kernel vector have a dimension of $(2 \cdot R_{max} + 1)(2 \cdot R_{max} + 1)$ and they represent the direction of the radial vector scaled by the distance from the center in each point. The convolution between the gradient versor images and these masks evaluates how many points in the image have a gradient direction concordant with the gradient direction of a range of circles. Then the peak in the accumulator array provides the center of the sub-image with higher circularity that is finally passed to the validation step. Examples of sub-images given as input to the ball recognition process are shown in figure 3.

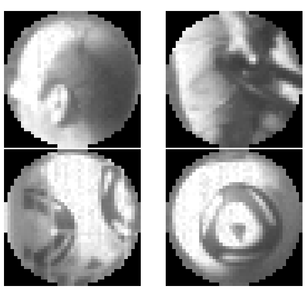


Figure 3. Some images of the training set: in the first row there are some negative examples of the ball, in the second row some positive examples

The validation step assesses the similarity of appearance between the candidate region and a set of positive examples stored previously. The similarity is evaluated by computing the Bhattacharyya distance among histograms (reduced to 64 bins): this measure is not computationally time consuming but at the same time it is sufficiently robust as it is invariant to the rotation of the target (textured balls are considered) and also to slight changes in scale. One of the strengths of the proposed system is that the construction and updating of the set of reference examples for validation takes place automatically.

Initially, the set of reference examples is empty and all the moving objects with the highest value of circularity (greater than a weak threshold) and with an area compatible with that of the ball are taken into account. Their displacement on the image plane frame after frame is then evaluated in order to estimate some motion parameters e.g. direction and velocity. The candidate regions are then included in the reference set if the associated motion parameters are compatible with those that only a ball can have in case of a shot on goal (direction towards the goal and plausible number of pixel displacement between frames). At the same time the relative distance into the image plane between the candidate regions and the other moving object in the scene is evaluated: if the relative distance is low and almost consistent, the ball candidate is discarded since it has likely been produced by incorrect segmentation of players' bodies.

The same criteria are used to add new examples in the reference set, additionally considering the value of the measurement of similarity with the pre-existing examples. The maximum number of examples in the reference set is 100 and it is managed as a circular buffer.

The reference set is re-initialized when circular objects with peculiar motion parameters and low similarity (i.e. higher distances in the space of histograms) to the examples in the reference set appear in a number of consecutive frames. This way the introduction of a new type of ball or sudden and substantial changes in lighting conditions (due to clouds or floodlights) can be handled automatically.

The ball has to be detected in more consecutive images in order to be sure that a true positive has been found. In this case, a different and more reliable procedure for selecting candidate moving regions is used (tracking phase). A ball position probability map, covering all the points of the processing image, is created as follows:

$$P(x,y) = \frac{e^{\left(-\frac{|(x - |\widetilde{x} + V_x sign(\cos\theta)|) + (y - |\widetilde{y} + V_y sign(\sin\theta)|)|^2}{2\sigma^2}\right)}}{\sigma\sqrt{2\pi}} \tag{5}$$

where $(\widetilde{x}, \widetilde{y})$ is the last known ball position and

$$\sigma = \frac{R_p V_{max} n}{R_{cm} T} \tag{6}$$

where V and θ are the local velocity and the direction of the ball in the image plane respectively, R_p is the Ball radius in pixels, R_{cm} is the Ball radius in centimeters and V_{max} is the maximum admissible velocity of the ball (in cm/sec), T is the camera frame rate and n is the number of frames between the past and actual ball detection (1 if, in this case, the two frames are consecutive). This way the maximum probability value is related to the point where, on the basis of past information about the ball's movement, the ball should be found (predicted point). The probability value decreases exponentially as the distance from the predicted point becomes close to 0 for points far from the last known ball position that cannot be reached considering the maximum speed limits (usually 120 km/h). In the following frames, the probability map is used to select candidate moving regions (like those with a probability greater than zero). This way, the ball can be detected both in case of merging with players and in case of partial occlusions. The ball velocity, direction and probability map are always updated using the proper value for n (i.e. the number of frames between the actual frame and the last ball detection). If the ball is not detected for three consecutive seconds (i.e. n becomes greater than T*3) the past information is considered outdated and the ball detection procedure starts again considering all the candidate ball regions in the whole image.

5. Supervisor node

The supervisor node has a decision-making function according to the processing results coming from the nodes. For each frame the processing units send several items of information to the supervisor, including the last frame number processed, the position of the ball (if detected), and the number of consecutive frames in which the ball has been correctly tracked. It should be noted that even if the images are synchronized in the acquisition process, the processing results are not necessarily synchronized, since each node works independently from the others. Moreover, a node may jump some frames having accumulated a significant delay during the processing. When the supervisor receives the results of three nodes for a given frame, or when it detects that synchronized data obtained cannot be retrieved for a given frame, the supervisor processes the obtained information to evaluate the occurrence of a goal event. This is done by evaluating the goal line crossing in the available 2D images. However, this way it is not possible to evaluate if the ball crossed the goal line inside the goal posts or not. For this reason, the 3D ball position and its trajectory before crossing the goal line are evaluated. This requires a calibration procedure (described in section **??**), and an accurate evaluation of the 3D position of the ball.

If the ball position is evaluated in the image plane, it is possible to estimate the corresponding projection line. The intersection of the three projection lines provides the estimate of the ball position in the real world coordinate system as shown in figure 4. In practice, this process entails uncertainty, so corresponding lines of sight may not meet in the scene. Furthermore, it is likely that in certain moments it is not possible to see the ball by one or more cameras because of occlusions, for example created by the players, the goalkeeper or the goalposts. For these reasons a special procedure for estimating the 3D position of the ball was introduced. If the ball is visible in only 2 cameras the 3D distance between the two projection lines is firstly computed. Then, if this distance is smaller than a selected threshold (typically about the size of the diameter of the ball, ie 22 cm.) the two projection lines are considered as referring to the same object (dealing with possible errors of the detection algorithms of the ball which are described in section 4) and then the mid-point of the common perpendicular to the two projection lines is chosen as an estimate of the 3D position of the ball. If the ball is visible in all three cameras, the mutual 3D distance among the three projection lines is calculated. The two projection lines with shorter distance are then considered the most reliable and this leads the calculation to the previous case. We are aware that different approaches have been proposed in literature to handle the 3D position estimation issue by triangulation ([9], [8]), but we have not considered using them because of the difficulties of their implementation and their high computational costs that make them unsuitable for a real-time system.

Finally, if the ball is only in a single camera, its 3D position can be estimated if some previous temporally close 3D positions are available. In this case, a linear filter is used to predict the next 3D position and then to estimate the projection lines of the missing views.

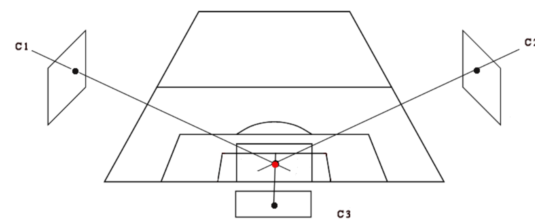


Figure 4. The intersection of the three projection lines produces the estimated ball position.

6. Experimental Results

A prototypal system was installed at the Friuli Stadium in Udine. The system uses Mikrotron MC1362 cameras with a spatial resolution of 1024x768 pixels at 504 fps and Xeon QuadCore E5606 2,13 Ghz as the processing node. Each node is equipped with a X64-CL iPro PCI frame grabber capable of acquiring images from one Medium Camera Link™ camera and performing image transfers at rates of up to 528 MB/s. The system was extensively tested during real "Serie A" championship matches and a specific experimental session (making use of the impact wall, slide, ball shooting machine, etc.) was conceived to evaluate goal event detection accuracy. Thus, both the system's reliability in the real context of a soccer match and its performance in very stressful sessions of shots executed using a ball shooting machine (which also allows repeatable results to be obtained) were tested.

An observation about experimental tests is mandatory: a comparison with other approaches is unfeasible, due to the complexity of the whole system. It could be interesting a comparison with commercial/industrial systems ([1], [2], [3]), but companies do not release technical information and data such as to make possible such comparisons.

6.1. Benchmark Results

Here we focus our attention on benchmark sessions performed outside of the match. In detail, four different experiments were carried out:

- Impact Wall shots (fig. 5(a) - 5(b)): during this test, an artificial wall was placed behind the goal line in two different positions: in the first, the ball position at the moment of impact is ”No Goal”, while in the second it is ”Goal” (ground truth). The ball was shot by a shooting machine at different speeds. This way the accuracy of the system to detect the goal event in terms of both spatial resolution and mostly temporal resolution (at high ball speed, even at 200 fps, there are just 1-2 frames to correctly detect the goal event) can be tested.

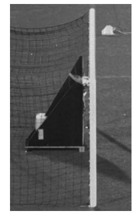
- Sled (fig. 5(c)): in this test, the ball is positioned on a mobile sled, and slowly moved from a non-goal to a goal position. The system’s precision to detect the exact ball position (in terms of cm over the goal line) was tested.

- Free Shots: during these experiments, several shots were performed by means of the shooting machine, in different positions with respect to the goal: left, right, middle, just under the horizontal crossbar, and just over the ground; each of them at different ball speeds. We tested whether the system fails to detect a specific portion of the goal area. Moreover, the reliability of the 3D reconstruction procedure (to separate shots inside and outside the goal posts) was tested.

- Outer Net (fig. 5(d)): in this session, specific shots on the outer net were performed with the final position of the ball inside the volumetric area of the goal, but arriving from outside the goal posts. We tested the system’s capability of tracking the ball accurately (even in 3D), by separating balls that arrived in the goal volumetric area from a ’goal trajectory’ (inside the goal posts) from balls that arrived from an external trajectory.

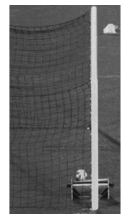
In order to show the weak impact of light conditions, some tests were also performed at night, in the presence of artificial stadium light. In table 1 the final results are reported. The *Impact Wall* sessions gave good results, with an overall success rate of more than 92%. The experiments were carried out by shooting the ball at different speeds, impacting the wall in different positions.

All shots in the *Outer Net* session were correctly detected; for the *Sled session*, we reported the mean distance over the goal line detected by the system, while a more detailed analysis of this data, to emphasize that the system mostly detected the event within 2 cm, is reported in table 2.



(a) Impact Wall Position for No Goal Simulation

(b) Impact Wall Position for Goal Simulation



(c) Sled

(d) Outer Net (note that the ball is over the net)

Figure 5. Example of different tests

The *Free Shots* session realized different results according to test lighting conditions: in the daylight test a success rate of over 98% was obtained. On the contrary, in the night test, an 88.54% success rate was obtained, in the same test. This was due to the different performances of the algorithms. First of all, the background subtraction together with computational aspects: if the segmentation algorithm, due to artificial light flickering, detects a number of moving points greater than reality, the following algorithms have to process more points causing a growing computational load, which leads to problems with memory buffers, and subsequently some frames are discarded (it should be noted that all our experiments were performed in realtime). To confirm this, this test session was off-line processed again, obtaining results comparable to those in daylight. It can be concluded that this drawback can easily be overcome, by simply optimizing the code and/or improving hardware with more performing components. The same observations are valid for the night session of the *Impact Wall* test.

Figure 6 reports images acquired during the experimental phase and corresponding to two goal events; the first row refers to a goal event during the Free Shot session, while in the second row images from the Impact Wall session are shown (just 2 cameras are reported for this experiment, the third one is evidently occluded and cannot help in any way).

6.2. Real match results

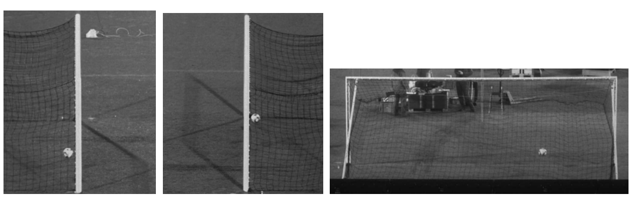
In order to evaluate the system’s robustness in real uncontrolled conditions, test were conducted during real matches of the Italian Serie A Championship; in particular,

Table 1. Overall performance of the GLT system during extensive experimental sessions.

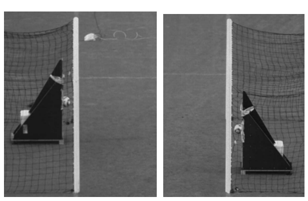
Test	Results
Impact Wall - Daylight	175/186 - 94.09%
Outer Net - Daylight	70/70 - 100%
Free Shots - Daylight	165/168 - 98.21%
Impact Wall - Night	84/93 - 90.32%
Free Shots - Night	85/96 - 88.54%
Sled - Daylight	average of 3.8 cm

Table 2. Sensibility evaluation of the system in the sled test.

Distance	Results
0 - 2 cm	17/32 - 53.125%
2 - 3 cm	8/32 - 25.00%
3 - 5 cm	4/32 - 12.50%
$>$ 5 cm	3/32 - 9.375%



(a) camera 1 (b) camera 2 (c) camera 3



(d) camera 1 (e) camera 2

Figure 6. Some images acquired during the experimental phase

the system was tested during 19 matches played at the Friuli Stadium in Udine (Italy). Table 3 reports the goal detection results. In a real context, the important events (goals) are limited in number so the benchmark sessions reported in the previous section are mandatory in order to test the system exhaustively. On the other hand, in a benchmark session, it is really hard to introduce and simulate all possible events that could happen during a real match: the presence of players in the scene that could alter the detection of the ball (some body parts, like legs and shoulders, could be erroneously detected as the ball); the presence of logos and/or particular combinations of colors on the players' uniforms that can influence the ball detection procedure; the possibility that players randomly cross the goal line (goalkeepers often do); the presence of objects in the goal area (towels, bottles, etc.) that could lead to misclassifications.

As it can be noted, during the 19 matches there were 33 goal events that were correctly detected (no misdetections) and just 1 false positive occurrence.

In figure 7, one of the goal events correctly detected (even if the ball was occluded by one camera and the ball appearance is not very different from the player's jersey) is shown. During this experimental phase, in addition to goal events, a very controversial situation occurred: the goalkeeper saved a shot when the ball was on the goal line (see fig. 8). The system evaluated that situation as No-goal and the image clearly evidences that it was right.

A false positive also occurred during a complex defensive action: four defenders were close to the goal line, trying to protect the goal and one of them kicked the ball away clearly before it crossed the line (see fig. 9). Afterwards, a defender crossed the goal line and, unfortunately, the system recognized the pattern of the ball on his shorts (whose position was also consistent with the trajectory predicted by the ball tracking procedure). Cases like this (although rare) could happen again, and certainly need further investigation. Considering that the system processed a huge amount of data, i.e a total of over 1.7K minutes of play, which correspond to about 20M of images, the percentage of errors can be considered acceptable.

Finally, something about computational load: a speedy response is mandatory for the system to actually be used. For this reason we evaluated the delay in response for each test. In fig. 10 a summary of the response time is reported. As evidenced, considering the realistic threshold of 2 seconds for the system's response, it can be noted that in about 80% of the total number of experiments the response time was acceptable. Considering that algorithms can be further improved and optimized, it can be concluded that the real-time constraint can easily be achieved.

Table 3. Performance in real conditions.

Goal Events	True Positives	False Negatives	False Positives
33 Goals	33	0	1

7. Acknowledgments

The authors thank Liborio Capozzo and Arturo Argentieri for technical support in the setup of the hardware used for data acquisition and processing.

References

[1] *http://en.wikipedia.org/wiki/Hawk-Eye*.

[2] *http://goalcontrol.de/gc4d.html*.

[3] *www.iis.fraunhofer.de/en/bf/ln/referenzprojekte/goalref.html*.

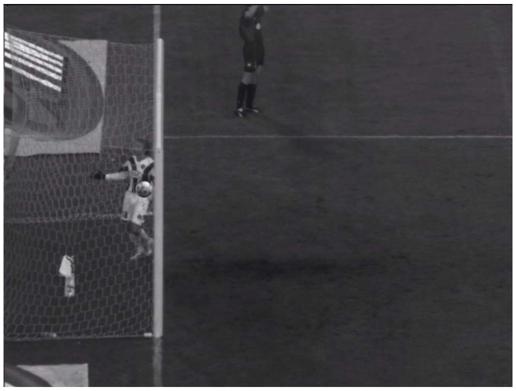


Figure 7. One of the 33 goal events occurred during the test on real matches.

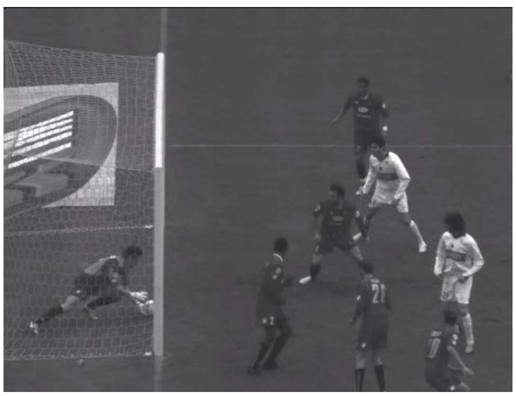


Figure 8. A controversial situation occurred during a real match and rightly evaluated by the system as no-goal

Figure 9. A situation in which the system erroneously detected a goal

[4] T. D'Orazio, M. Leo, P. Spagnolo, P. Mazzeo, M. Nitti, N. Mosca, and A. Distante. An investigation into the feasibility of real-time soccer offside detection from a multiple camera system. *IEEE Transaction on Circuits and Systems for Video Technology*, 19(12):1804 – 1817, 2009.

[5] T. D'Orazio, M. Leo, P. Spagnolo, M. Nitti, N. Mosca, and A. Distante. A visual system for real time detection of goal events during soccer matches. *Computer Vision and Image Understanding*, 113(5):622 – 632, 2009.

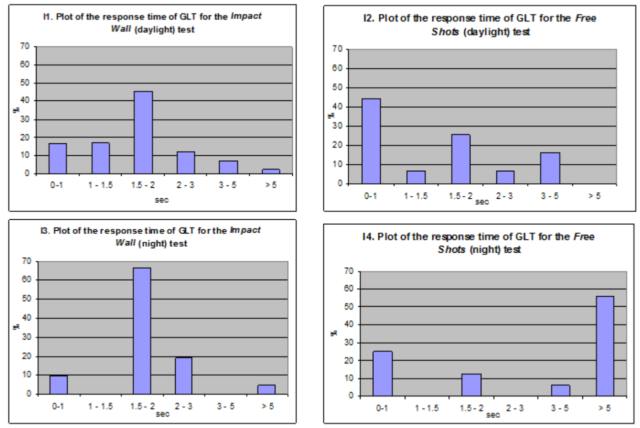


Figure 10. Plot of the response time

[6] A. M. Fischler and R. C. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM*, 24(6):381–395, 1981.

[7] S. Hashimoto and S. Ozawa. A system for automatic judgment of offsides in soccer games. In *IEEE International Conference on Multimedia and Expo*, pages 1889–1892, 2006.

[8] K. Kanatani, Y. Sugaya, and H. Niitsuma. Triangulation from two views revisited: Hartley-sturm vs. optimal correction. In *BMVC*, 2008.

[9] P. Lindstrom. Triangulation made easy. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 1554 –1561, june 2010.

[10] Y. Liu, D. Liang, Q. Huang, and W. Gao. Extracting 3d information from broadcast soccervideo. *Image and Vision Computing*, 24(10):1146–1162, 2006.

[11] V. Pallavi, J. Mukherjee, A. Majumdar, and S. Sural. Ball detection from broadcast soccervideos using static and dynamic features. *Journal Visual Communication and Image Representation*, 19(7):426–436, 2008.

[12] M. Piccardi. Background subtraction techniques: a review. In *IEEE SMC 2004 International Conference on Systems, Man and Cybernetics*, 2004.

[13] J. Ren, J. Orwell, G. Jones, and M. Xu. Real-time modeling of 3-d soccer ball trajectories from multiple fixed cameras. *IEEE Transaction on Circuits and Systems for Video Technology*, 18(3):350–362, 2008.

[14] P. Spagnolo, N. Mosca, M. Nitti, and A. Distante. An unsupervised approach for segmentation and clustering of soccer players. In *IMVIP Conference*, pages 133–142, 2007.

[15] X. Yu, H. Leong, C. Xu, and Q. Tian. Trajectory-based ball detection and tracking in broadcast soccervideo. *IEEE Transaction on Multimedia*, 8(6):1164–1178, 2006.