A visual system for real time detection of goal events during soccer matches

T. D'Orazio, M. Leo, P. Spagnolo, M. Nitti, N. Mosca, A. Distante
Institute of Intelligent Systems for Automation - C.N.R.
Via Amendola 122/D-I, 70126 Bari, ITALY
E-mail: dorazio@ba.issia.cnr.it

Abstract
During soccer matches a number of doubtful situations arise that cannot be easily judged by the referee committee. An automatic visual system that checks objectively image sequences would prevent wrong interpretations due to perspective errors, occlusions, or high velocity of the events. In this work we present a real time visual system for goal detection. Four cameras with high frame rates (200 f/s) are placed on the two sides of the goal lines. Four computers process the images acquired by the cameras detecting the ball position in real time; the processing result is sent to a central supervisor which evaluates the goal event probability and, when the goal is detected, forwards a warning signal to the referee that takes the final decision.

1 Introduction
An automatic system for goal detection would be appreciated by the referee committee as a decision support tool during soccer matches in order to prevent wrong interpretation due to perspective errors, occlusions, and high velocity of events. A positioning-tracking system that is under experimentation at the moment relies on tiny microwave emitters placed inside the ball, the players' shin guards and the goal structures. These emitters broadcast signals including sensor identifiers at high frequency; eight antennas receive the signals. A position estimation system determines the position of each microwave emitter through triangulation. The position estimate is typically within five to eight cm of the actual position. The main drawback of this kind of technology is that it cannot provide an objective check of detected events. If a decision has to be taken by the referee board it must be settled on an objective evaluation of the considered event.

In this work we present a visual system able to detect the goal event through real time processing of the acquired images and to provide also an objective analysis of the goal event showing the image sequence that records the event under consideration.

1.1 Related Work
The analysis of sports video data has received great attention in the last few years but the main interests have been concentrated on automatic highlight detection, since finding the clips of highlights manually from large amounts of video data is a boring and tedious task. Many researchers have published papers about soccer video analysis and goal event detection. They are based on the
observation that the most significant events are generally followed by slow motion replay. Some
special patterns of cinematic features can be used to locate goals or other important events. In [4]
a goal event is detected if a break in the game is recognized during which the emotions of one
or more players are captured, and several slow motion replays from different cameras are shown.
The analysis of dominant colors together with motion intensity has been used in [6] to locate the
clips containing soccer field images and separate two mutually exclusive states of the game, play
and break. In [1] the grass-ratio, which measures the amount of grass pixels in a frame, is used
to classify the main shots, whereas the peaks in the Hough Transform have been used to detect
the goal area and classify near-goal actions. In [2] both time-domain and frequency-domain audio
features are learned during a training phase to construct a decision tree and recognizes goal shot
actions. In a similar way, in [5] goal events are detected using camera motion information as visual
cue and the loudness as an audio descriptor. In [3], a set of color and motion features are provided
to Dynamic Bayesian Networks to separate play and break phases in soccer video. Audio visual
features and a Support Vector Machine have been used in [7] to identify common characteristics
of successful scoring events across different genres of sports videos.

Despite a lot of research efforts for soccer summarization in broadcast video, real time analysis of
soccer images for particular event detection hasn’t received great attention in literature. The high
velocity of soccer events imposes strict real time constraints that usually make the summarization
algorithm for these applications inapplicable. Furthermore the detection of high velocity events
requires cameras with higher frame rates than the broadcast ones. In these cases the processing
times have to be very short to follow the real time events.

1.2 Our Contribution

Automatic ghost goal detection is an unsolved problem which is getting particular attention from
referee associations, sport press and supporters. A ghost goal occurs when, after a shot, the ball
touches the under side of the cross bar, bounces back into the field of play having completely
crossed the goal-line without touching the net. Usually these kinds of shots reach high velocity
and often the official visual abilities related to their position on the field are not enough to observe
the event. In this paper we present a real time visual system for goal detection during soccer
matches, which can be used as decision support by the referee committee. Different points were
considered to define the main characteristics of the visual system. First of all it was necessary to
decide on the number and the positions of the cameras for observing the goal event; the frame rate
and the resolution were properly selected in order to assure the registration of high velocity shots.
Then real time algorithms, for detecting the ball in the image sequences provided by each camera,
were devised. Finally a decision making supervisor had to evaluate the goal event probability
according to the information provided by the processing from each camera.

The rest of the paper is organized as follows: Section 2 describes the system in detail; section
3 describes the ball detection algorithm; section 4 presents the decision making supervisor; the
results of the tests executed over the course of two seasons are presented in Section 5. In Section
6 conclusions are presented.
2 Outline of the Proposed System

In order to devise a visual system for goal detection a number of points had to be considered both for the hardware involved and for the software requirements. The limitations of the referees to detect this kind of events are basically due to perspective errors, and human limitations of image processing. In fact, the chance of seeing a ball that touches the under side of the cross bar and immediately bounces back into the field of play having completely crossed the goal-line, is guaranteed only if the observation points are on the plane of the goal line. Furthermore, because of the high velocity of these shots, the human visual system can be limited in observing the events. Also the cameras used for broadcast images that have a resolution of $30 \text{fr/sec}$ could not register the events. In figure 1 the ball displacements between consecutive frames that can be recorded with different frame-rates as a function of the shot velocity are plotted. Considering a possible maximum ball velocity of $120 \text{Km/h}$ it is necessary to use cameras with $200 \text{fr/s}$ in order to observe a displacement of $20 \text{cm}$. In figure 2 and 3 we show an image sequence of a goal event acquired with a high frame rate camera ($200\text{fr/sec}$) and a standard broadcast camera($30\text{fr/sec}$). In the first case the goal is clearly visible in at least three images of the sequence (the images in which the whole of the ball passes over the goal line), whereas in the second case the goal is not visible since there isn’t a frame in which the ball is completely over the goal line. This first consideration imposes strict constraints on the hardware involved, i.e. the camera technology, the memory requirements, the processors, the recording capability, etc, but also on the processing algorithms that have to manage a very large image flow.

![Object Velocity and Displacement Relationship](image)

Figure 1: The relation between the displacement and the velocity of the ball with different frame rates

In figure 4 the visual system is outlined. Four cameras have been placed on the four sides of the goals, with the optical axes parallel to the goal line plane. Each camera is connected to a processor that records the images and processes them. The four processors are connected to a main node, which has the supervisor function. It evaluates and compares the processing results of the relevant nodes (two for each side of the soccer field) and provide an alarm signal if the goal event is
Figure 2: The image sequence of a ghost goal event acquired with a high frame rate camera. The goal is clearly visible in at least three images of the sequence (in the last row).

recognized. In the next two sections we describe the processing executed by each processing node and the supervisor node.

3 Processing Nodes

In figure 5 a schematic diagram of the processing steps executed by each node is shown.

3.1 Initial Calibration

First of all it is necessary to do a calibration step during which the correspondence between a set of four points on the ground plane and the image plane is assessed. This step is necessary at the beginning of the match for the evaluation of the homography. In this way it is possible to evaluate the transformation matrix $M$ that relates the points in the image plane with the corresponding points in the real plane. If the ball position is evaluated in the image plane it is possible to estimate the viewing line as shown in figure 6. The intersection of the two opposite viewing lines provides the estimate of the ball position in the real world coordinate system.

We have selected four points that are visible to the two opposing cameras and we have measured in a real coordinate system their positions and the positions of the two cameras ($C_1, C_2$). The
Figure 3: The image sequence of a ghost goal event acquired with a low frame rate camera. The goal is not recorded.

Figure 4: The scheme of the visual system

general projective transformation from one projective plane to another is represented as

\[
\begin{bmatrix}
  x_1 \\
  x_2 \\
  x_3
\end{bmatrix} =
\begin{bmatrix}
  m_{11} & m_{12} & m_{13} \\
  m_{21} & m_{22} & m_{33} \\
  m_{31} & m_{32} & m_{33}
\end{bmatrix}
\begin{bmatrix}
  X_1 \\
  X_2 \\
  X_3
\end{bmatrix}
\]  

(1)

or \( x = MX \).

If the transformation is represented in Cartesian coordinates it becomes:

\[
\begin{align*}
  u &= \frac{x_1}{x_3} = \frac{m_{11}X + m_{12}Y + m_{13}}{m_{31}X + m_{32}Y + m_{33}}, \\
  v &= \frac{x_2}{x_3} = \frac{m_{21}X + m_{22}Y + m_{23}}{m_{31}X + m_{32}Y + m_{33}}
\end{align*}
\]  

(2)

where \((u, v)\) are the coordinates in the image plane and \((X, Y)\) are the coordinates in the real plane. The correspondence between the \((u, v)\) coordinates and the \((X, Y)\) coordinates of four points allows the solution of the equation system that provides the matrix \(M\). Repeating the evaluation for the two cameras starting from the position of the ball in the two image plane, it makes possible to obtain the corresponding projections \((P_1, P_2)\) on the real plane as shown in figure 6. The lines passing through \((C_1, P_1)\) and \((C_2, P_2)\) will intersect at the real position of the ball in the soccer field. Then, in order to apply this homographic transformation, it is necessary to know the 3D real coordinates of the two cameras and also the real coordinates of the four reference points and the corresponding image coordinates on the two image planes. This calibration needs to be done only once, after the camera installation, and if the cameras remain fixed these measures are still valid for any match.
3.2 Moving Object Segmentation

The segmentation of the image to extract moving objects is fundamental as it limits the ball search to small areas and reduces the computational time. Additionally, the analysis of connected regions allows the evaluation of the size of each region and so the possibility of discarding large areas in which the ball search need not be applied since they are not compatible with the ball dimension.

For this purpose a segmentation algorithm has been implemented. Using static cameras allows the detection of moving regions by with the use of a background subtraction algorithm. The procedure consists of a number of steps. At the beginning of the image acquisition a background model has to be generated and later continuously updated to include lighting variations in the model. Then, a background subtraction algorithm distinguishes moving points from static ones. Finally, a connected components analysis detects the blobs in the image.

The implemented algorithm uses the mean and standard deviation to give a statistical model of the background. Formally, for each frame the algorithm evaluates:

\[
\mu^t(x, y) = \alpha \mu^{t-1}(x, y) + (1 - \alpha) \mu^{t-1}(x, y) \tag{3}
\]

\[
\sigma^t(x, y) = \alpha |\mu^t(x, y) - \mu^t(x, y)| + (1 - \alpha) \sigma^{t-1}(x, y) \tag{4}
\]
It should be noted that 4 is not the correct statistical evaluation of standard deviation, but it represents a good approximation of it, allowing a simpler and faster incremental algorithm which works in real time. The background model described above is the starting point of the motion detection step. The current image is compared to the reference model, and points that differ from the model by at least two times the correspondent standard deviation are marked. Formally, the resulting motion image can be described as:

\[ M(x, y) = \begin{cases} 
1 & \text{if} \quad |I(x, y) - \mu^t(x, y)| > 2 \cdot \sigma^t(x, y) \\
0 & \text{otherwise} 
\end{cases} \]  

where \( M(x, y) \) is the binary output of the subtraction procedure. An updating procedure is necessary to have a consistent reference image at each frame a requirement of all motion detection approaches based on background. The particular context of application imposed some constraints. First of all, it is necessary to quickly adapt the model to the variations of light conditions, which can rapidly and significantly modify the reference image, especially in cases of natural illumination. In addition, it is necessary to avoid including in the background model players who remain in the same position for a certain period of time (goalkeepers are a particular problem for goal detection as they can remain relatively still when play is elsewhere on the pitch). To obtain these two opposite requirements, we have chosen to use two different values for \( \alpha \) in the updating equations (3) (4). The binary mask \( M(x, y) \) allows us to switch between these two values, and permits us to quickly update static points (\( M(x, y) = 0 \)) and to slowly update moving ones (\( M(x, y) = 1 \)). Let \( \alpha_S \) and \( \alpha_D \) be the two updating values for static and dynamic points respectively:

\[ \alpha(x, y) = \begin{cases} 
\alpha_S & \text{if} \quad M(x, y) = 1 \\
\alpha_D & \text{otherwise} 
\end{cases} \]  

In our experiments we used \( \alpha_S = 0.02 \) and \( \alpha_D = 0.5 \). The choice of a small value for \( \alpha_S \) is owed to the consideration that very sudden changes in light conditions can produce artifacts in the binary mask 5: in such cases these artifacts will be slowly absorbed into the background, while they would remain permanent if we had used \( \alpha_S = 0 \).

The binary image of moving points, the output of the background subtraction step, has to be analyzed to extract the regions of interest for the ball searching process. A connectivity analysis has been implemented which scans the entire image, groups and labels neighboring pixels into regions. For each pixel, the 8-neighboring pixels are considered and put together in a list until no more connected points are found. The search is repeated until all the moving points have been considered. After this step we know for each moving region the position in the image and the number of points that belongs to it. In this way, regions with an area less than a given threshold are considered as noise and removed, whilst large regions, that probably correspond to players, are not processed by the ball search procedure.

### 3.3 Ball Detection

An automatic method that detects the ball position in each image is the central step to building the vision system. In the soccer domain a great number of problems have to be managed, such as occlusions, shadowing, misdetection (wrong detection of objects similar to the ball), and lastly 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 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 [8–11]; example based techniques can be applied to learn the salient features of a class of objects from sets of positive and negative examples [13,14].

The main contribution of our work is in using two different techniques together in order to take advantage of the peculiarity of each of them: 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 for containing the ball; second, a neural classifier evaluates all the information contained inside the selected area to validate the ball hypothesis. The circle detection algorithm is based on the Circle Hough Transform (CHT) which has been formulated as convolutions applied to the edge magnitude image [17]. The convolution kernels have been properly defined in order to detect the most completed circle in the image, being independent of the edge magnitude, and according to different shapes of the ball and to different light conditions. The sub-image containing the result of the detection process is forwarded to a neural network classifier trained to recognize "ball" and "no-ball" instances. In order to reduce the input data to the classifier, but still maintain the information properties that characterize the ball, different preprocessing phases were applied to the selected area and the behaviors of the obtained classifiers compared on the same test images. A large number of experiments were carried out on real image sequences. Also varying light conditions were considered in order to demonstrate the robustness of the ball recognition system.

3.3.1 Circle Detection

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. In this way only an arc perpendicular to the edge orientation at a distance R from the edge point needs to be plotted. The CHT and also its modifications can be formulated as convolutions applied to an edge magnitude image (after a suitable edge detection) [17]. We have defined a circle detection operator that is applied over all the image pixels, produces a maximal value when a circle is detected with a radius in the range \([R_{\text{min}}, R_{\text{max}}]\):

\[
u(x,y) = \frac{\int \int_{\mathcal{D}(x,y)} \bar{c}(\alpha, \beta) \cdot \bar{O}(\alpha-x, \beta-y) d\alpha d\beta}{2\pi (R_{\text{max}} - R_{\text{min}})}
\]

where the domain \(\mathcal{D}(x,y)\) is defined as:

\[
\mathcal{D}(x,y) = \{ (\alpha, \beta) \in \mathbb{R}^2 | R_{\text{min}}^2 \leq (\alpha-x)^2 + (\beta-y)^2 \leq R_{\text{max}}^2 \}
\]

\(\bar{c}\) is the normalized gradient vector:

\[
\bar{c}(x,y) = \left[ \frac{E_x(x,y)}{|E|}, \frac{E_y(x,y)}{|E|} \right]^T
\]

and \(\bar{O}\) is the kernel vector

\[
\bar{O}(x,y) = \left[ \frac{\cos(\tan^{-1}(y/x))}{\sqrt{x^2 + y^2}}, \frac{\sin(\tan^{-1}(y/x))}{\sqrt{x^2 + y^2}} \right]^T
\]
The use of the normalized gradient vector in (9) 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 could be 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 for ensuring we have the same values in the accumulation space when circles with different radii in the admissible range are found. Moreover the 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 last 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 in each point the direction of the radial vector scaled by the distance from the center. The convolution between the gradient versor images and these masks evaluates how many points in the image have the gradient direction concordant with the gradient direction of a range of circles. Then the peak in the accumulator array gives the candidate center of the circle in the image.

### 3.3.2 Ball Recognition

The sub-image containing the result of the detection process is passed to a neural classifier trained to separate images as ”ball” or ”no-ball” instances. The neural network, trained with a backpropagation algorithm, consists of three layers of nodes: the number of nodes of input layer depends on the number of coefficients selected after the image preprocessing (major details will be given in the rest of this section); the hidden layer has 80 nodes and the output layer has only one node. The neural network parameters have been found experimentally, choosing the combination that produces the best results in terms of ball detection rate.

![Figure 7](image_url)

Figure 7: Some images of the training set: in the first row there are some negative examples, in the second row some positive examples of the ball.

Examples of positive and negative sub-images given as input to the ball recognition process are shown in figure 7. In the second row of figures there are some images of the ball after applying the background subtraction procedure. It transpires that the information of the ball concept is encapsulated both in texture and contour: there is a region with some texture inside, which has a sharp separation from the background. It is necessary to discover if a more compact representation of the image can be used to store the main part of the image information in a small set of coefficients, still preserving the data characterizing the problem and taking into account the
different appearances of the searched pattern with varying light conditions. In this work a Discrete Wavelet Transform is used, supplying a hierarchical representation of the image implemented with the iterative application of two filters: a low pass filter LPF (approximation) and a high pass filter HPF (detail filter) [15, 16]. At each step the Wavelet Transform breaks the image into four sub-sampled images (sub-images), applying first to rows and then to columns the LP HP filtering scheme, followed by a decimator of factor 2. This procedure is iterated several times to obtain a full depth wavelet decomposition, re-applying the same scheme only on the low pass sub-image at each step.

In figure 8a) the sub-image arrangement scheme of a 3-level Wavelet transform is shown. The capital letters in each sub-image represent the kind of filters that have been applied on the image of the previous level; the first letter is the filter that has been applied in the horizontal direction, while the second letter is the filter that has been applied in the vertical direction (H stands for a High Pass Filter L stands for a Low Pass Filter).

![Figure 8: a) The sub-image arrangement scheme of the decomposition with a 3-level Wavelet Transform; b) The 3-level Wavelet Transform on a sub-image containing the Ball](image)

The band LL is a coarser approximation of the original image. The bands LH and HL record the changes of the image in horizontal and vertical directions. The band HH shows the high frequency components of the image. Decompositions can be iterated on the LL sub-bands. After applying a 3-level Wavelet Transform, an image is decomposed into sub-bands of different frequency components. Figure 8b) shows the result of the application of a three stage wavelet decomposition on the ball image. The coefficients of the wavelet transform hold information about the texture and the shape of the object in the image. In this way it is possible to distinguish the ball from other elements that could have one of the two aspects in common. For example the head or the back of a player could have the same shape and size of the ball but it can be distinguished by considering the information on texture. In the first row of figure 7 there were some no-ball examples that have been erroneously detected in the first circle detection step. However, a proper analysis of the texture information allows the classifier to distinguish them from the ball examples. Different tests were carried out in order to detect the best decomposition level, and the filter most suitable for the problem at hand. Additionally, different dimensions of the images were considered in order to have the highest detection rate of the neural classifier. For greater details on the experimental results of the ball detection step see [17].
3.4 Ball Tracking

Usually the ball searching procedure is applied after the connectivity analysis only to those regions identified as candidates to contain the ball based on position and dimensions. Large regions are not considered since they are assumed to contain players. Regions that are behind the goal line are not analyzed to avoid the detection of events that are not significant. In fact in a goal event the ball has to be seen somewhere on the pitch in front of the goal line and then to pass over the goal line. As soon as the ball detection algorithm recognizes the ball in the image a tracking procedure is initiated based on a time-space coherence of the ball 2D trajectory. During the tracking phase the connectivity analysis is no longer used and the ball search is applied to all the moving regions of the image without any consideration of their position. For this reason in each successive image the ball search procedure is limited to an area around the estimated ball position in order to accelerate the process. Taking into account the ball velocity (speed and direction), the ball search is applied firstly around the predicted ball position then, if the ball is not found, to an enlarged window around the predicted position. In this way we firstly seek the presence of the ball assuming a linear trajectory, and immediately after we seek the presence of the ball in any point around the last position assuming a deviation in trajectory. The scale of enlargement of the search window has been estimated by considering the maximum displacement of the ball and the frame rate acquisition parameter. If, in the successive image, the ball is not found the searching window is further enlarged by the same fixed quantity. This enlarging process is iterated until either the ball is found in the image or the window dimension reaches the maximum image dimension. In the first case the ball search returns to a limited area, while in the second case, after a fixed number of frames, the tracking process is stopped and the normal ball searching procedure is switched on.

4 Supervisor node

The supervisor node has a decision making function according to the processing results coming from each pair of cameras. The strategy is based on some heuristics that make data fusion evaluating the time space coherence of the ball 3D trajectory. The processing results of the two corresponding nodes are compared and a probability function is evaluated. 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), the neural network response on the filtered image after the wavelet transform ($NN_1$), the neural network response on the original grey level image ($NN_2$), and the number of consecutive frames in which the ball has been correctly tracked.

It should be noted that the processing results are not necessarily synchronized since each node works independently from the others. Moreover a node may jump some frames having collected a significant delay during the processing. When the supervisor receives the results of two nodes for a given frame, or when it detects that a pair of synchronized views cannot be obtained for a given frame (and hence it must consider a single view) the supervisor processes the obtained information to detect the goal event and evaluates the associated goal probability.

The supervisor node analyzes the information to decide from the following possibilities: Goal, No Goal, probable Goal, Line Passing. For each pair of opposite nodes the supervisor evaluates the node answer.

\[
\text{NodeAnswer}^t = \begin{cases} 
\text{Ball} & (NN_1^{t-2} \text{or } NN_2^{t-2}) \text{and } (NN_1^{t-1} \text{or } NN_2^{t-1}) \text{and } (NN_1^t \text{or } NN_2^t) \\
\text{NoBall} & \text{(otherwise)}
\end{cases}
\]
The best evaluation is obtained when two synchronized frames (or within two/three consecutive frames) are processed by the two opposite nodes, each providing a positive answer. But even in this case there are chances that the result obtained by one camera doesn’t concur with the result obtained by the opposite camera. In order to assess this point the homographic control is taken into consideration, and hence it is possible to observe if the ball position detected by a processing unit concurs with the ball position detected by the other processing unit. This is done by considering the minimum distance between the two viewing lines.

\[
\text{HomogControl} = \begin{cases} 
\text{Ball} & (\text{MinDist(line}_{t_1}(C_1, P_1), \text{line}_{t_2}(C_2, P_2)) < \text{thr}) \land (|t_1 - t_2| < 3f) \\
\text{No Ball} & \text{(otherwise)} 
\end{cases}
\]

The lines passing through \((C_1, P_1)\) at the time \(t_1\) and \((C_2, P_2)\) at the time \(t_2\) are those shown in figure 6. If the minimum distance goes beyond a fixed threshold, the supervisor detects that the results of the two processing units are not compatible and therefore has to invalidate the result of the weaker processing unit. This is accomplished by comparing the uncertainty index of the two views and considering the least uncertain view as the more reliable. The uncertainty index is evaluated for each node by considering the difference between the prediction of the ball position and the actual position estimated.

The last point that the supervisor has to consider is position of the ball with respect to the Goal Plane. Using the two viewing lines of the two cameras, it is possible to determine the 3D ball position (BallPos). To detect a goal event, the ball position must be identified as beyond the goal line and inside the goalposts. However, due to the fact that the homographic transformations are usually affected by small errors (which can modify the decision inside/outside the woodwork when the ball is very near to the crossbar or one of the goalposts) an uncertainty region has been defined in which the system cannot be sure about the ball position.

\[
\text{PositionControl} = \begin{cases} 
\text{Inside} & \text{(BallPos} \in [lgp + t, rgp - t]) \\
\text{Probably Inside} & \text{(BallPos} \in [lgp - t, lgp + t] \land \text{BallPos} \notin [lgp + t, rgp - t])} \\
\text{Outside} & \text{(otherwise)} 
\end{cases}
\]

where \(lgp\) is the left goal post position, \(rgp\) is the right goal post position and \(t\) is the homographic threshold.

In table 1 the summary of the possible supervisor decisions is shown. The best results are obtained when two views are available. If the ball position detected by one node concurs with the ball position detected by the other node, the homographic control asserts that both the views are observing the same object (that corresponds to the Ball) and according to the Position Control results the system decides if a Goal, No Goal or a Probable Goal has occurred. In particular if the Goal is validated not immediately, but after a few frames, and there is evidence that the ball was found soon after the goal line by one camera, the supervisor node is able to signal the goal event in the first visible frame. If the Homographic control does not validate the Ball it means that one of the two nodes is observing something other than the ball, and in this case as with the last two rows of the table, the system has just one view and can confirm that the ball has passed the goal line, but without knowing if it is inside or outside the goal.

5 Experimental Results

The software has been implemented by using Visual C++ on a Dual Processor XEON 3.2 Ghz chipset 7525 and a CORECO frame grabber model X64-IPROCL. The cameras are JAI TM6740CL.
Table 1: The summary of the supervisor decisions

<table>
<thead>
<tr>
<th>System Decision</th>
<th>Node 1 Answer</th>
<th>Node 2 Answer</th>
<th>Homographic Control</th>
<th>Position Control</th>
</tr>
</thead>
<tbody>
<tr>
<td>Goal</td>
<td>Ball</td>
<td>Ball</td>
<td>Ball</td>
<td>Inside</td>
</tr>
<tr>
<td>No Goal</td>
<td>Ball</td>
<td>Ball</td>
<td>Ball</td>
<td>Outside</td>
</tr>
<tr>
<td>Probable Goal</td>
<td>Ball</td>
<td>Ball</td>
<td>Ball</td>
<td>Probably Inside</td>
</tr>
<tr>
<td>Line Passing</td>
<td>Ball</td>
<td>No Ball</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>Line Passing</td>
<td>No Ball</td>
<td>Ball</td>
<td>—</td>
<td>—</td>
</tr>
</tbody>
</table>

with 640×480 image size that acquire 200f/s. The focal length has been chosen to have the ball dimension of 20x20 pixels. Due to the reduced field of view of the camera, most of the time the image is empty. For this reason we decided to process normally 1 frame over 20 frames. When the ball enters in the camera view the processing is activated on all the frames acquired. The processing time of each frame depends on the complexity of the scene. The maximum processing time for the whole image is 8ms, that can produce a delay that becomes longer as the action duration increases. During our experiments we recorded an average delay of 2 seconds. These results are very encouraging since not all the code optimization have been completed and no specialized hardware has been used.

A range of experiments have been undertaken to evaluate the whole system. A total of 2434 minutes were recorded and processed in real-time during 27 soccer matches over the course of two seasons. The test set of matches covered a range of weather conditions (including rain and fog) and different kick off times (afternoon, evening with artificial lighting). The system response was continually observed by a human operator, to confirm the system results. However, the real-time nature of events and the operator physical limitations (i.e. being able only to view one perspective at a time), limits the efficacy of this evaluation and it was necessary to further compare the system results with the ground truth determined post-hoc. To facilitate this task, image sequences were manually labelled to represent the ground-truth.

For each match, goal events were identified by the system, and subsequently saved as individual clips at the end of the match. Other actions, including some that did not contain the ball at all, were selected manually and saved along with the other clips, to add some noise to the test set. Each identified piece of action, including the clips selected manually, was taken by two cameras that saved independents but synchronized clips. At the end of the experimentation phase, each extracted clip were evaluated manually by a human operator. This was done to record a reference response, i.e. what is actually happening in that clip (the ground truth), including the event type, and the ball position frame by frame. An ideal system must give the same response.

The action in the clip is classified using the following event types:

**Goal** A goal occurs in the clip. Since the goalposts are fully contained in the field of view of each camera assigned to them, a goal is always seen by two cameras.

**Beyond** The ball has crossed the goal-line somewhere. It is possible that the ball, during the exact frame in which it crosses entirely the goal-line, is seen by one or both cameras.
**No Event** This category covers all the remaining situations in which the ball enters in the camera view without crossing the goal line.

Finally, the user labels the ball position frame by frame. Although for a goal-line detection system, the only important information lies in the frames where the ball has crossed the goal-line, we have chosen to label the entire collection of clips to include information that is needed to evaluate the system. It is important to decide when to and when not to label the position. This is made considering two factors:

- The particular ball detection implemented in the system logically defines a region of interest on the image where the ball should be successfully recognized (near the goal) and certain regions of the image (after and on either side of the goalpost) are therefore ignored. This speeds up the processing phase.

- The particular test methodology implemented uses a reference response which is the first frame when the ball has totally crossed the goal-line. This frame is named the "reference goal-line frame", and is used to evaluate the delay in the system to recognize when the ball has crossed the goal-line (or the goalpost). However, when labelling we need to bear in mind the danger of perspective error.

A frame is therefore labelled if it contains the ball and the ball is less than 25% occluded by any object.

When the labelling phase has ended, the batch processing phase begins and the system response for each clip is logged. The clips are split using their semantic classification (Goal, Beyond, No Event) and then processed using a custom test program that computes the experimental results. The test set contains 208 synchronized sequences, for a total of 416 individual film segments. The test sequences have different time lengths, but usually are short clips of 3000 frames (15 seconds at 200fps). The longest sequences are three times this length.

### 5.1 Test 1

The first test (Table 2 and Table 3) evaluates the performance of each camera, and therefore each clip, individually. Since the system is based on an imaging solution, there is a small chance that its performance is slightly influenced by the goalpost and camera position with respect to the stadium shape and orientation (both affecting the way that light and shadows are projected onto the playing field, and which vary throughout the course of the day). The first table evaluates the average delay, in number of frames, from the reference goal-line frame, when the real cross-line event occurs, to the frame in which the system detects the goal-line event. The performance of each camera is evaluated by checking the correspondence between the reference response and the system response. The goal clips and the beyond clips produce two distinct tables (Table 2 and Table 3 respectively). The tables show, in the rows, the response delays for each camera. In the first column the number of clips in which each node has seen the goal event in the first correct frame (match on the first valid frame) is shown, followed in the second column by the number of clips in which the detected event has a delay of one frame with respect to the real event (match on the second frame), and so on, ending with the number of matches on the tenth frame. If the distance between the event detected and the real event passes the ten-mark, the event is labelled as "Not seen" and is shown in the last column of the table. It’s worth noting that the comparisons
Table 2: Nodes evaluation on Goal clips: on each row the number of clips in which the node detects the ball is reported. The columns represent the frame position in which the ball has been detected for the first time beyond the reference goal-line frame.

<table>
<thead>
<tr>
<th>Frame position beyond the reference goal-line frame</th>
<th>Number of processed clips for each node</th>
</tr>
</thead>
<tbody>
<tr>
<td>1° 2° 3° 4° 5° 6° 7° 8° 9° 10° Not seen</td>
<td></td>
</tr>
<tr>
<td>Node 1</td>
<td>12  2  0  0  0  0  0  0  0  0  0</td>
</tr>
<tr>
<td>Node 4</td>
<td>10  3  1  0  0  0  0  0  0  0  0</td>
</tr>
<tr>
<td>Node 3</td>
<td>25  2  1  1  0  0  0  0  0  0  0</td>
</tr>
<tr>
<td>Node 2</td>
<td>24  2  1  1  0  0  1  0  0  0  0</td>
</tr>
</tbody>
</table>

Table 3: Nodes evaluation on Beyond clips: on each row the number of clips in which the node detects the ball is reported. The columns represent the frame position in which the ball has been detected for the first time beyond the reference goal-line frame.

<table>
<thead>
<tr>
<th>Frame position beyond the reference goal line frame</th>
<th>Number of processed clips for each node</th>
</tr>
</thead>
<tbody>
<tr>
<td>1° 2° 3° 4° 5° 6° 7° 8° 9° 10° Not seen</td>
<td></td>
</tr>
<tr>
<td>Node 1</td>
<td>36  1  1  1  0  1  0  0  0  1  4</td>
</tr>
<tr>
<td>Node 4</td>
<td>32  5  2  1  1  0  1  0  0  0  7</td>
</tr>
<tr>
<td>Node 3</td>
<td>39  4  1  2  0  0  0  0  0  0  6</td>
</tr>
<tr>
<td>Node 2</td>
<td>32  5  3  0  1  1  2  0  0  0  10</td>
</tr>
</tbody>
</table>

are made subsequently, beginning with the reference goal frame and are stopped at the first match, so the sum of the columns equals the number of clips related to that camera. Therefore, if each column in a row (for a camera) is divided by the total number of clips (for that camera), a delay probability distribution function can be obtained.

Looking at the results of the first test when running on goal sequences (Table 2), it is possible to note that most of the goal events are detected in the first valid frame (that is, in the same goal frame signalled by the user during the labelling phase stating the ground truth). All the goal events are seen within the first 10 valid frames (and nearly all within the first 5 valid frames). It should be noted that these evaluations are limited to the nodes and cannot be used to evaluate the exact frame in which goal events are signaled by the system. In fact the supervisor node merges the data coming from the two opposite nodes and, by evaluating the ball 3D trajectory, is able to signal the goal events in the first valid frames. For example, there is a clip in which the Node 2 detects the ball for the first time in the seventh frame beyond the reference goal-line frame because in the preceding frames the ball was occluded by one player. In this case, however, the supervisor validates the event in that frame, but it signals the goal in the first frame in which the opposite camera sees the ball.

The results of the first test for the sequences marked "Beyond" are shown in Table 3. Most of the events are detected within the first valid frames. Differently from the Goal clips, in this case we have a number of undetected events ("Not seen"). This is due to the fact that, in general,
Table 4: System evaluation on "Beyond", "Goal", "NoEvent" clips

<table>
<thead>
<tr>
<th>Ground truth</th>
<th>System response</th>
<th>Number of Processed clips</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Goal</td>
<td>No Goal</td>
</tr>
<tr>
<td>Goal</td>
<td>38</td>
<td>0</td>
</tr>
<tr>
<td>No Event</td>
<td>0</td>
<td>19</td>
</tr>
<tr>
<td>Beyond</td>
<td>0</td>
<td>42</td>
</tr>
<tr>
<td>Beyond - 1 View</td>
<td>0</td>
<td>9</td>
</tr>
<tr>
<td>Beyond - 2 Views</td>
<td>0</td>
<td>33</td>
</tr>
</tbody>
</table>

the neural networks, used to recognize the ball, were trained using patterns extracted from balls sampled at the center of the scene, in the proximity of the goalposts. Balls that are far from the goalposts are not in the depth of field of the cameras and have a different appearance from those used in the training set.

5.2 Test 2

The primary task of the second test is to evaluate the response of the system as a whole. While the first test looked at the response of each individual node, the second test evaluates the system response once the processing done on the nodes is integrated through the supervisor which merges the different responses and takes a final decision. The system responses to the test sequences are reported in table 4. The ground truth of the clips, manually classified as "Goal", "NoEvent" and "Beyond", is reported in the rows of the table. In particular, in the last two rows "Beyond" clips have been divided according to the number of views that see the ball beyond the goal line (1 View, 2 Views). The system response, reported in the columns, is classified in "Goal", "No Goal", "Probable Goal", "Line Passing". The last column reports the number of the sequences tested for each user category.

Data integration augments the system performance. The first row shows the system response on the 43 goal clips. It can be observed that 38 events are detected from both views and therefore identified as goal. Five events are identified as "Probable Goal": they are seen from both the two views, but the resulting homography measure is contained in a "limbo" zone which is near the goalpost and cannot be correctly identified as inside or outside the posts due to the homography accuracy permitted by the camera positioning.

The second row (No Event clips) shows very promising results. The clips manually chosen to test the system included sequences featuring crowded scenes, players in the goalpost, scenes with various objects in the goalposts, none of which caused the system evaluation to fail.

The third row shows the system results on the "Beyond" events. It can be noted that the vast majority of them are signaled as Line Passing. Many of them are actually seen from only one camera, while the others are due, as pointed out before, to the different ball appearance between the goal and beyond events. Four clips have been signaled as Probable Goal since the Position Control reported them as belonging to the uncertainty region (the ball passes very near to the goalpost).

In figures 9, 10, 11, and 12 different responses of the system are shown. In figure 9 the system correctly recognizes a goal event in the first visible frame when the two views of the ball are
available. In figure 10 the system signals the goal event in the first valid frame even if the ball is occluded in one view by the goalkeeper’s glove. This is due to the capability of the supervisor to reconstruct the ball trajectory having the ball 3D positions in some frames before and after the occlusion. In figure 11 a Line Passing event is detected by the system since the ball is visible from only one view. A No goal event is shown in the figure 12. It has been correctly detected by the system that evaluates the ball 3D position using the homographic and position control and decides that the ball is outside the goalpost.

Figure 9: A goal event has been correctly detected in the first visible frame when the two views of the ball are available.

Figure 10: A goal event has been correctly detected in the first valid frame even if the ball is occluded in one view.

Figure 11: A Line passing event has been correctly detected since the ball is visible from only one view.
6 Discussion and Conclusions

Aim of this work was to develop a real time visual system for goal line monitoring that could respect the main constraints imposed by the International Football Federation (FIFA) for the usage of technologies during official matches: first of all to be not invasive for the field and the players, to be completely automatic and reliable, to be not dependent on human intervention, and to have real time responses. In this paper we present the results obtained from the experiments carried out during the championship 2005−2006/2006−2007, with four cameras with high frame rates (200f/s) placed on the two sides of the goal lines. Four computers processed the images acquired by the cameras, detecting the ball position in real time. The processing result was sent to a central supervisor, which evaluated the goal event probability and took a decision of Goal, No Goal, Probable Goal or Line Passing. The proposed system can send a warning signal of line passing only to the referee, who is free to decide if he should accept the suggestion or follows what he believes to be a better choice.

The system was tested during 27 official soccer matches for a total of 2434 minutes of processing, during which the real time performance of the system was tested. Different off-line tests were executed on a selection of significant video clips in which the ground truth was manually identified by a human operator. The tests demonstrate that the system is always able to recognize goal events in the very first frames in which they are visible. During these experiments 38 out of 43 goal clips were detected as goal events and the remaining as Probable goal. Besides no false positive was generated by the system during the experimentation over the course of two seasons. However we cannot exclude that the system could fail if the combination of particular pattern configurations and ball trajectory occurs. If the ball were correctly tracked and the goalkeeper or one player (wearing something with a texture similar to the ball texture) passed the goal line hiding the ball from both the cameras, the ball recognition procedure would give a false positive on something with a circular shape (on the shoulder, the gloves, the shorts). Four Beyond clips have been detected as Probable goal since the ball passed very near the goalpost. The insertion of a third camera behind the door could provide many advantages to the system [18]. Multiple triangulations could cut off both the Probable Goal responses, eliminating the uncertainty region in which the system cannot be sure about the ball position, and solve possible ball occlusions.

Current work is focused on the code optimization in order to further reduce the processing time and decrease the system latency when a goal event occurs at the end of a very long and complex action.
References


[18] PCT/IT02/00039 registered by CNR-ISSIA, *System and method for the measurement of the relative position of an object with respect to a point of reference* International Publication number WO 07/061684 A2