A Robot Referee for Robot Soccer*

Matías Arenas, Javier Ruiz-del-Solar, Simón Norambuena, and Sebastián Cubillos

Department of Electrical Engineering, Universidad de Chile
{marenas,jruizd}@ing.uchile.cl

Abstract. The aim of this paper is to propose a robot referee for robot soccer. This idea is implemented using a service robot that moves along one of the field sides, uses its own cameras to analyze the game, and communicates its decisions to the human spectators using speech, and to the robot players using wireless communication. The robot uses a video-based game analysis toolbox that is able to analyze the actions at up to 20fps. This toolbox includes robots, ball, landmarks, and lines detection and tracking, as well as refereeing decision-making. This robot system is validated and characterized in real game situations with humanoid robot players.

1 Introduction

One of the RoboCup main goals is allowing robots to play soccer as humans do. A natural extension of this idea is having robots that can referee soccer games. Refereeing task are very similar to playing task, but differentiate in the fact that a referee has to correctly interpret every situation, a single wrong interpretation can have a large effect in the game result. The main duty of a robot referee should be the analysis of the game, and the real-time refereeing decision making (referee decisions can not be delayed). A robot referee should be able to *follow the game*, i.e. to be near the most important game actions, as human referees do. In addition it should be able to communicate its decisions to the human or robot players, assistant referees, and spectators. This communication can be achieved using speech, gestures, data networks or visual displays, depending on the distance of the message's receptor, and the available communications mechanisms. The robot referee should primarily use its own visual sensors to analyze the game. In large fields or in games where the ball moves very fast or travels long distances, the robot referee could use external cameras, in addition to assistant referees. Thus, a robot referee should have 3 main subsystems: (i) video-based game analysis, (ii) self-positioning and motion control, and (iii) interfaces to communicate decisions. Interestingly, the video-based game analysis subsystem, in addition to be used for refereeing decision-making, can be used to obtain game statistics (% of the time that the ball is in each half of the field, % of ball possession of each team, number of goals of each team player, etc.), as well as for video annotation and indexing, which could be later used to retrieve an automated summary or a semantic description of the game. Besides, the robot referee could be used as commentator of robot soccer games.

* This research was partially supported by FONDECYT (Chile) under Project Number 1061158.

L. Iocchi et al. (Eds.): RoboCup 2008, LNAI 5399, pp. 426 – 438, 2009.
© Springer-Verlag Berlin Heidelberg 2009

In this context, the aim of this paper is to propose a robot referee for robot soccer. This robot referee is specially intended to be used in the RoboCup SPL 2-legged league, and in the RoboCup humanoid league. The referee is a service robot that moves alongside one of the field lines (see figure 1 and 2), uses its own cameras to analyze the game, and communicates its decisions to the human spectators using speech and to the robot players using wireless communication. The robot video-based game analysis subsystem is based on the one proposed in [27], but the robot detection module has been largely improved by the use of statistical classifiers. One interesting feature of the robot referee is its ability to express facial gestures while refereeing, like being angry when a foul is committed or happy when a goal is scored, which makes him very attractive to human spectators. To the best of our knowledge a similar system has not been proposed in the literature. This paper is organized as follows. In section 2 some related work is presented. The here-proposed robot referee is presented in section 3. In section 4 some experimental results of the application of this system are presented. Finally, in section 5 conclusions of this work are given.

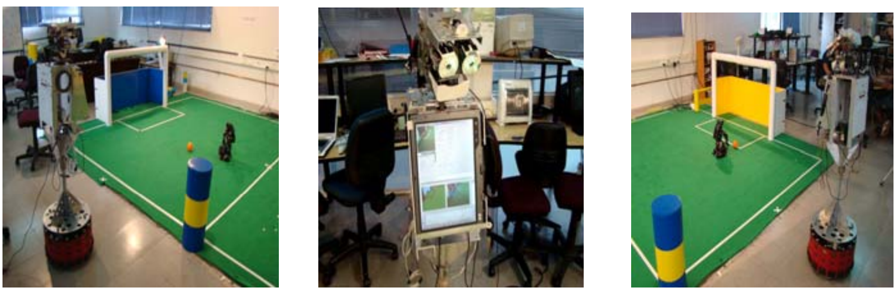

Fig. 1. The robot referee in a typical game situation

2 Related Work

Computer vision based analysis of sport videos has been addressed by many authors (e.g. [1]-[21]), and nowadays is a hot topic within the multimedia video analysis community. There are also successful video analysis programs that are being used by TV sport channels (e.g. Hawkeye [22] and Questec [23]). Applications have been developed in almost all massive sports such as tennis ([11][12][14][18]), soccer ([2][4][5][15][16][20][21]), baseball ([8][13]), basketball ([10]), and American football ([9]). However, to the best of our knowledge the automatic analysis of robot sports has not been addressed in the literature, except for [27].

The major research issues for the automatic analysis of human sports include (see a survey in [18]): ball and players tracking, landmarks detection (goal mouth, oval, side lines, corners, etc.), tactic analysis to provide training assistance, highlight extraction (ace events in tennis, score events in soccer and basketball, etc.), video summarization (automatic generation of game summaries), content insertion (e.g. commercial banner replacement in the field), and computer-assisted refereeing (e.g. offside detection). Some of these research issues are still open as for example fully autonomous tactic analysis, soccer offside detection considering player's intention, or video summarization; current solutions are semi-automatic and provide assistance to human referees or video operators. We believe that many of the accumulated knowledge in the automatic analysis of human sports can be employed for the automatic analysis of robot sports. Probably the main obvious difference being that in the robot sport case, robot identification is a hard problem, because usually all players look the same, because they all correspond to the same robot model (e.g. RoboCup SPL league). However, the robots can be individualized using the team uniform color [24] and the player's number [26].

The here-proposed robot referee makes use of the accumulated knowledge in automatic analysis of human sports, and in the RoboCup soccer leagues. The robot video-based game analysis subsystem is based on [27], however, the robot detection (one of the core referee functionality) is improved, instead of using SIFT descriptors, a cascade of boosted classifiers allows the robust detection of robots.

3 Proposed Robot Referee

3.1 Hardware

As robot referee we use our service robot *Bender*. This personal/social robot was originally designed to be used for the RoboCup @Home league. The main idea behind its design was to have an open and flexible testing platform. The robot has shown to be adequate for the @Home league (it won the *RoboCup @Home 2007 Innovation Award*), and has been used to provide multimedial and ubiquitous Web interfaces [29], and as speaker (lecturer) in activities with children (see pictures in [34]). Bender's main hardware components are:

- A robot's chest that incorporates a tablet PC as the main processing platform of the robot. We use a HP TC4200, powered with a 1.86 GHz Intel Centrino, with 1 GB DDR II 400 MHz, running Windows XP Tablet PC edition. The tablet includes 802.11bg connectivity. The screen of the tablet PC allows: (i) the visualization of relevant information for the user (a web browser, images, videos, etc.), and (ii) entering data thanks to the touch-screen capability.
- A robot's head with pan and tilt movements, two CCD cameras, one microphone and two loudspeakers. The head incorporates a face with the capability of expressing emotions. The head is managed by a dedicated hardware, which is controlled from the tablet PC via USB.
- A robot's arm with 3 degrees of freedom (DOF), two in the shoulder and one in the elbow. The arm is powered with a 3-finger hand. Each finger has 2 DOF. The arm is managed by a dedicated hardware, which is controlled from the tablet PC via USB.
- A mobile platform where all described structures are mounted. The platform provides mobility (differential drive), and sensing skills (16 infrared, 16 ultrasound, and 16 bumpers). The whole platform is managed by a dedicated hardware, which is controlled from the tablet PC via USB.

One interesting feature of Bender is that the relative angle between the mobile platform and the robot body can be manually adjusted. In the robot standard configuration

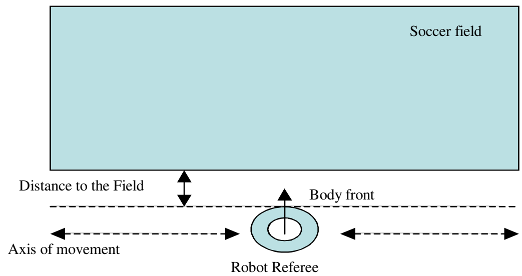

Fig. 2. Robot referee positioning

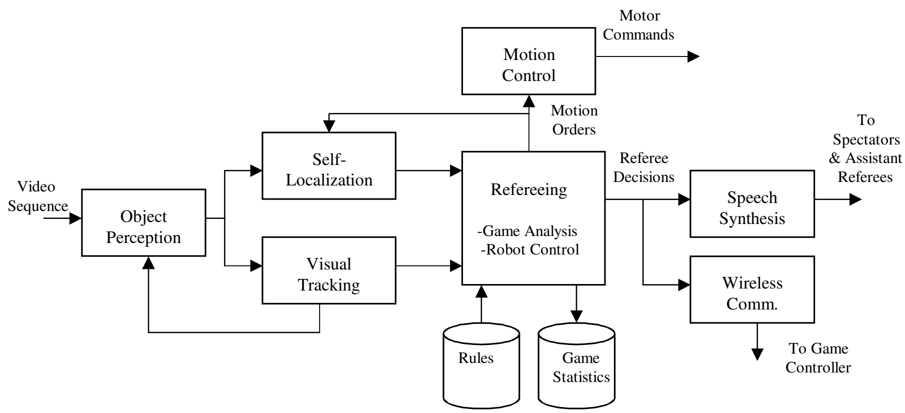

Fig. 3. Block diagram of the robot referee controller

this angle is set to 0 degrees, which allows the normal robot movement. For the task of refereeing, the angle is set to 90 degrees. This allows the robot to have a frontal view of the field while moving along the line, even though it has a differential drive configuration (see figure 2).

3.2 Robot Controller

The block diagram of the proposed robot referee controller is shown in figure 3. The system is composed by seven main modules *Object Perception*, *Visual Tracking*, *Self-localization*, *Refereeing*, *Motion Control*, *Speech Synthesis*, and *Wireless Communications*, and makes use of two databases: *Rules* (input) and *Game Statistics* (output).

The *Object Perception* module has two main functions: object detection and object identification. First, all the objects of interest for the soccer game (field carpet, field and goal lines, goals, beacons, robot players, ball) are detected using color segmentation and some simple rules, similar to the ones employed in any RoboCup soccer robot controller. No external objects, as for example, spectators or legs of assistant referees or team members are detected (in some leagues assistant referees and team members can manipulate the robots during a game). The identification (identity

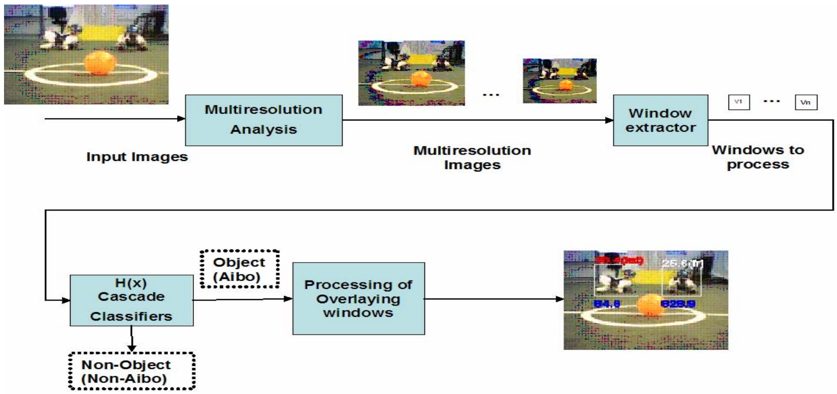

Fig. 4. Block diagram of the detection system

determination) of goals, beacons and the ball is straightforward, because each of them has a defined form and color composition. The identification of field and goal lines is carried using the relative distance from the detected lines to the robot referee, and to the already identified beacons and goals. The detection of the robot players is more difficult, and it is performed using a multiscale robot detection framework based on the use of boosted classifiers, as proposed in [28]. This multiscale robot detection framework (see block diagram in figure 4) works as follows: (i) To detect the robots at different scales, a multiresolution analysis of the images is performed, by down-scaling the input image by a fixed scaling factor --e.g. 1.2-- (*Multiresolution Analysis* module). This scaling is performed until images of about 24x24 pixels are obtained. (ii) Windows of 24x24 pixels are extracted in the *Window Extraction* module for each of the scaled versions of the input image. (iii) The windows are analyzed by a nested cascade of boosted classifier (*Cascade Classification Module*). (iv) In the *Overlapping Detection Processing* module, the windows classified as positive (they contain a robot) are fused (normally a robot will be detected at different scales and positions) to obtain the final size and position of the detections. Real-time robot processing is achieved thanks to the use of cascades of classifiers, and because this analysis is carried out only at the beginning of each robot tracking sequence (see the feedback connection from *Visual Tracking* to *Object Perception* in figure 3).

The *Visual Tracking* module is in charge of tracking the moving objects, i.e. the ball and the robot players. The implemented tracking system is built using the mean shift algorithm [30], applied over the original image (not the segmented one). The seeds of the tracking process are the detected ball and robot players. As in [32], a Kalman Filter is employed to maintain an actualized feature model for mean shift. In addition, a fast and robust line's tracking system was implemented (see description in [27]). Using this system, it is not necessary to detect the lines in each frame. Using the described perception and tracking processes, the system is able to track in near real time (up to 20 fps) all game moving objects and the lines.

The *Self-localization* module is in charge of localizing the robot referee. As in the case of the robot players, this functionality is achieved using the pose of the landmarks (goals and beacons) and the lines, and odometric information. The only difference being that in the case of the robot referee, the movements are not executed inside the field, but outside, along one of the field sides (see figure 2).

The *Refereeing* module is in charge of analyzing the game dynamics and the actions performed by the players (e.g. kicking or passing), and detecting game relevant events (goal, ball out of the field, illegal defender, etc.). This analysis is carried out using information about static and moving detected objects, and the game rules, which are retrieved from the *Rules* database. In addition, this module is in charge of the referee positioning. The module should keep the referee outside of the field, but at a constant distance of the field side (the referee should move along one of the field sides), it should control de robot's head and body movement to allow the robot to correctly *follow the game*, by avoiding obstacles and without leaving the field area (in case that the ball or a player leave the field.). It is important that the referee always perceives and follows the main elements of game-play. In the present this is done by following the ball, and estimating the position of the players in the field. In future implementations we plan to use several cameras to have more information of the activities in the field. The outputs of this module are refereeing decisions (e.g. goal was scored by team A) that are sent to the *Speech Synthesis* and *Wireless Communication* modules, motion orders that are sent to the *Motion Control* module, and game statistics (e.g. player 2 from team A score a goal) that are stored in the corresponding database.

The *Motion Control* module is in charge of translating motion orders into commands for the robot motors. These commands allow the control of the robot pose, the robot head pose, the robot facial expressions, and the robot arm.

Finally, the *Speech Synthesis* and *Wireless Communication* modules communicate the referee decisions to robot players and human assistant referees and spectators. Wireless communication is straightforward, while speech synthesis is achieved using the CSLU toolkit [35].

3.3 Refereeing

This module is in charge of analyzing the game dynamics, determining the actions performed by the players, and the game relevant events. This analysis is carried out using the information of the static and moving objects detected, and the game rules, which depend on the specific RoboCup soccer leagues.

Most of the game situations can be analyzed using information on the position of the ball and the robot players in the field, the time during which the ball and the robot players stay in a given field area, the localization of the field lines and goal lines, and for some specific cases, the identity of the last robot that touches the ball. For instance, to detect a goal event, a ball crossing one of the two goal lines should be detected. The correct positioning of a given robot depends on its own position and in some cases the position of the other players and the ball. The identity of the scoring robot is the identity of the last robot that touches the ball. Thus, using simple rules many situations such as "goal detection", "ball leaving the field", "game stuck", "robot kickoff positioning", "robot leaving the field" or "robot falling" can be detected. There are complex situations that depend on the exact relative position between two players or between a player and the ball that cannot be robustly detected. Some exemplar situations are "ball holding", "goalie/player pushing" and "robot obstruction".

For example, in the case of the RoboCup humanoid league, from which rules will be used to run our experiments, the following situations (definitions taken from [31]) can be analyzed:

- *Goal*: “A goal is scored when the whole ball passes over the goal line, between the goal posts and under the crossbar, provided that no infringement of the rules has been committed previously by the team scoring the goal”.
- *Robots kickoff positioning*: “All players are in their own half of the field. The opponents of the team taking the kick-off are outside the center circle until the ball is in play. The ball is stationary on the center mark. The referee gives a signal. The ball is in play when it is touched or 10 seconds elapsed after the signal.
- *Ball In and Out Play*: “The ball is out of play when it has wholly crossed the goal line or touch line whether on the ground or in the air or when play has been stopped by the referee. The ball is in play at all other times, including when it rebounds from a goalpost, crossbar, corner pole, or human and remains in the field of play”.
- *Global Game Stuck*: “the referee may call a game-stuck situation if there is no progress of the game for 60s”.
- *Illegal defense and attack*: “Not more than one robot of each team is allowed to be inside the goal or the goal area at any time. If more than one robot of the defending team is inside its goal or goal area for more than 10s, this will be considered illegal defense. If more than one robot of the attacking team is inside the opponent's goal or goal area for more than 10s, this will be considered illegal attack”.

However, there are some other situations that are much harder to analyze, because it is required either to judge the intention of the players (e.g. robot pushing) or to solve visual occlusions that difficult the determination of the relative position of the robot legs or the ball (e.g. ball holding). For the moment we have not implemented the detection of those situations.

The game statistics that can be computed in our current implementation are: % of the time that the ball is in each half of the field, number of goals of each team and team player, number of direct kicks to the goal by each team and each team player, number of times that each team and each team player sent the ball out of the field, number of illegal defense or attack events of each team, number of times that each team player leaves the field, number of times that each player fall down, number of global game stuck events, and time required for automatic kickoff positioning by each team.

3.4 Robot Perceptor

To detect the playing robots a cascade of boosted classifier was used (based on Adaboost). This detector requires a diversified database of images with the robots that are going to be detected, as well as a number of images not containing any robots. The referee here implemented was mainly focused on the RoboCup Humanoid league, so the training database was made from videos obtained in the league website (videos submitted by the league’s teams) and using our humanoid robot (Hajime H18) in our laboratory.

During the training of the cascades, validation and training sets are used. The procedure to obtain both sets is analogous, so only the training dataset is explained. To obtain the training set used at each layer of the cascade classifier, two types of databases are needed: one of cropped windows of positive examples (Humanoid) and one of images not containing the object to be detected. The second type of database is used during the bootstrap procedure to obtain the negative examples (this comes from our implementation of the Adaboost algorithm). The training dataset is used to train the weak classifiers, and the validation database is used to decide when to stop the training of a given layer and to select the bias values of the layer (see details in [28]). To obtain positive examples (cropped windows) a rectangle bounding the robot was annotated and a square of size equal to the largest size of the rectangle was cropped and downscaled to 24x24 pixels. The "positive" database contains several thousand 24x24 images containing Humanoids in different poses, environments, illuminations, etc. The training process was repeated several times in order to obtain a good detection rate with a low quantity of false positives. Each time the process was repeated more images were added to the database to increase the variety of the images included in it (so the classifier becomes more general). The number of images used in the final version of the Robot detector is shown in Table 1.

4 Experimental Results

To prove the usefulness of the proposed system, some experiments were carried out. The results of these tests are shown in the following sections. The quantitative results were obtained from a series of video sequences (with different configurations) taken in our laboratory. In total 5,293 frames were taken for a preliminary analysis of our system. These videos are from short play sequences in the field, some examples can be seen in figures 5, 6 and 7. The idea was to capture different playing sequences, but due to the huge amount of possible situations during a game, only a few of them were captured.

4.1 Object Detection and Tracking

The first evaluation was the Robot Detector module. To do this, the system was programmed to detect robots every 5 to 10 frames, and to track the detections in every frame. Then we counted all the robots that the detector correctly found versus the number of robots that appear on all the frames. We also counted the false detections that appeared in some frames. The tracking systems are much faster than the detection systems, especially the mean shift module used to track the detected robots. This allows the system to work close to real time and the possibility to run faster or slower depending on the possible applications and desired results. Table 2 shows the robot detection results. These results show that the robots were correctly detected in almost all frames, with a false detection every 15 or so frames although, usually, false detections appeared in consecutive frames. These results are quite good and show that the detector is working as intended. Once the robot is correctly detected the tracking system works remarkably well (the mean shift system has no problem tracking object in these environments).

Table 1. Summary of the databases used for training

Class	# Positive examples		# Negative images	
	(Training)	(Validation)	(Training)	(Validation)
Humanoids	3,693	3,693	6,807	3,410

Table 2. Summary of the results for the Robot Detector module

Number of Frames	Number of Robots	Robot Detection Rate	Number of false positive windows
5,293	3,405	98.7%	334

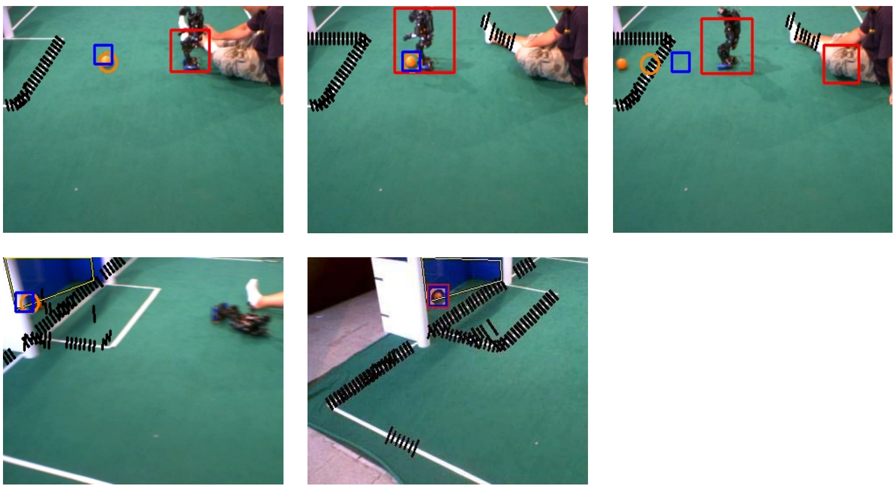

Fig. 5. Selected frames from a robot scoring sequence. Robot/ball tracking window in red/blue. The goal detection event is shown by a red window out of the ball blue window.

Line detection was also evaluated, but only partially. Many lines of many sizes appear in all images and they can appear entirely, partially or barely. The "correct" detection of all lines is difficult to put in numbers because of this, and the many criteria that can exist (a "correctly detected" line can mean that a part of the line was detected or that some parts of the line were detected or that all the line was detected, etc.). Our line detector worked fine on side lines that appeared entirely (or almost), having problems only with those that were farther away. It also detected some of the lines inside the field but had problems with smaller lines (the ones that limit the space for the goalie). Some examples can be seen in figures 5 and 6. The back lines were detected in almost all cases, while the goal area of the lines was a bit more difficult to tag correctly. Our system does a good job at tagging a zone in front of the goal but can sometimes over- or under-estimate the portion of the line that represents the goal line. Beacons, goals and ball detection could be evaluated, but these systems have already been tested before (they are very similar to the ones currently used by the

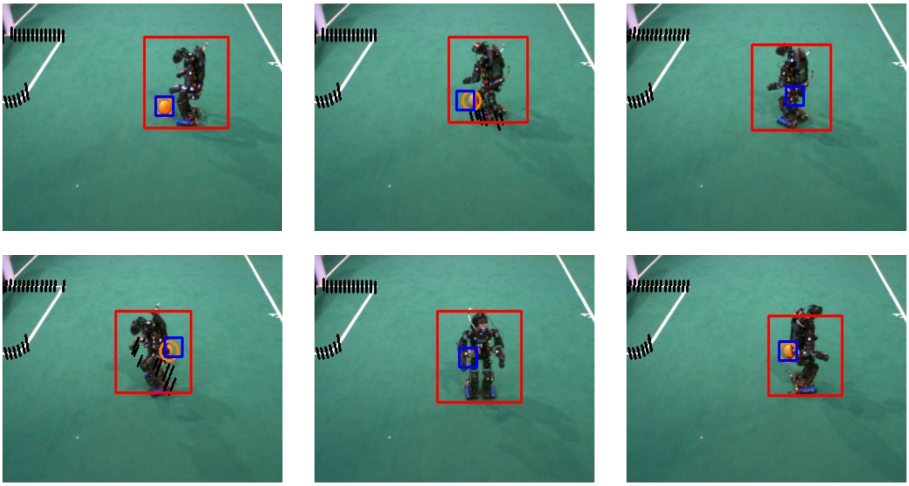

Fig. 6. Selected frames from ball tracking and robot detection

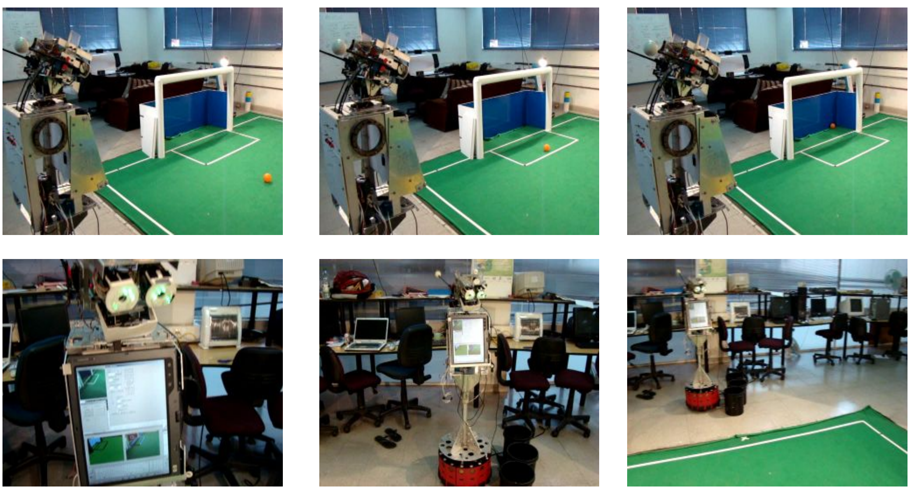

Fig. 7. Selected images of a Refereeing situation

robots in RoboCup league), and give very good results. An example of this is figure 6 which shows the ball detection system and how it works even with total occlusion of the ball (thanks to a Kalman filter).

4.2 Event Detection

Same video sequences as in 4.1 were employed. As these events are less frequent than object apparitions, a statistical evaluation is more complicated. Furthermore these events are harder to detect because a correct detection of all the elements present in the event must be present. For example for goal detection the system needs a good detection on the lines, goal, goal line and ball. Figures 5 and 7 show goal detections, one directly from the camera used by the robot, and the other from an external camera filming the setup of the refereeing robot. Our system correctly detects most of the goal situations, although some false positives appear in some cases (where our goal line is over-estimated and the ball was leaving the field instead of entering the goal). Other situations like "game stuck" or "player leaving the field" where correctly detected but they have to undergo more real games testing to see if it is really working as intended. "Illegal defense and attack" situation still need more work, mainly because of the difficulties of setting the goalie area with line detections.

5 Conclusions

This paper proposes a robot referee for robot soccer. A custom robot is used alongside the soccer field to referee, analyze and possibly comment the current game. It is currently implemented for the RoboCup Humanoid and SPL leagues, but it can be extended to other robot soccer categories. The system employs a custom robot that can move along the soccer field and follow the action by watching the game with a camera mounted in its head. The robot uses a conventional computer mounted in its chest to process the images from the game and analyze its content. The computer also controls the robot moving parts in order to follow the game and to notify the surrounding persons the important events ongoing in the game (goal, ball leaving the field, etc).

Currently, the developed system can detect and identify all humanoid-league defined field objects, and perform the tracking of the moving objects in real-time. A good part of the defined situations can be correctly detected, and other show good results but need more testing in diverse environments while some other need more work to function as intended. The proposed system has shown great potential but needs to be refined for more complex situations. Furthermore it needs testing in different environments to completely prove its usefulness.

The system could be criticized because is it not assured that it always take the right decision (e.g. due to occlusion problems), but human referees also not always take correct decisions. This is especially true in robot soccer environments, where very often non-experimented humans assume refereeing tasks. Future implementations plan to use more cameras to cover the whole field, and to be able to correctly evaluate more complex situations. These cameras can be fixed over the field or in another moving robot along the other side of the field. In most collective sports more than one referee is used to make sure all rules are obeyed, so it is logical to include more sources of information in this case too.

References

1. Assfalg, J., Bertini, M., Colombo, C., Del Bimbo, A.: Semantic Annotation of Sports Videos. IEEE MultiMedia 9(2), 52–60 (2002)
2. Assfalg, J., Bertini, M., Colombo, C., Del Bimbo, A., Nunziati, W.: Semantic Annotation of soccer videos: automatic highlights identification. Computer Vision and Image Understanding 92(2-3), 285–305 (2003)

3. Babaguchi, N., Kawai, Y., Kitahashi, T.: Event Based Video Indexing by Intermodal Collaboration. IEEE Transactions on Multimedia 4(1), 68–75 (2002)
4. Bertini, M., Del Bimbo, A., Nunziati, W.: Soccer Video Highlight Prediction and Annotation in Real Time. In: Roli, F., Vitulano, S. (eds.) ICIAP 2005. LNCS, vol. 3617, pp. 637–644. Springer, Heidelberg (2005)
5. D'Orazio, T., Ancona, N., Cicirelli, G., Nitti, M.: A ball detection algorithm for real soccer image sequences. In: Proc. Int. Conf. Patt. Recog. – ICPR 2002, Canada, August 11-15 (2002)
6. Duan, L.Y., Xu, M., Chua, T.S., Tian, Q., Xu, C.S.: A mid-level representation framework for semantic sports video analysis. In: Proc. of ACM MM 2003, Berkeley, USA, November 2-8, pp. 33–44 (2003)
7. Ekin, A., Tekalp, A.M.: Automatic soccer video analysis and summarization. IEEE Trans. on Image Processing 12(7), 796–807 (2003)
8. Han, M., Hua, W., Xu, W., Gong, Y.H.: An integrated baseball digest system using maximum entropy method. In: Proc. of ACM MM 2002, pp. 347–350 (2002)
9. Li, B., Sezan, M.I.: Event detection and summarization in American football broadcast video. In: Proc. SPIE Conf. on Storage and Retrieval for Media Databases, vol. 4676, pp. 202–213 (2002)
10. Nepal, S., Srinivasan, U., Reynolds, G.: Automatic detection of 'Goal' segments in basketball videos. In: Proc. ACM MM 2001, Ottawa, Canada, pp. 261–269 (2001)
11. Pingali, G., Jean, Y., Carlbom, I.: Real time tracking for enhanced tennis broadcasts. In: Proc. IEEE Conf. Comp. Vision and Patt. Rec. – CVPR 1998, pp. 260–265 (1998)
12. Pingali, G., Opalach, A., Jean, Y.: Ball tracking and virtual replays for innovative tennis broadcasts. In: Proc. Int. Conf. Patt. Recog. – ICPR 2000, Barcelona, Spain, pp. 4146–4152 (2000)
13. Rui, Y., Gupta, A., Acero, A.: Automatically extracting highlights for TV Baseball programs. In: Proc. of ACM MM 2000, pp. 105–115 (2000)
14. Sudhir, G., Lee, J., Jain, A.K.: Automatic classification of tennis video for high-level contentbased retrieval. In: Proc. of IEEE Int. Workshop on Content-based Access of Image and Video Database, pp. 81–90 (1998)
15. Tovinkere, V., Qian, R.J.: Detecting semantic events in soccer games: Towards a complete solution. In: Proc. ICME 2001, pp. 1040–1043 (2001)
16. Wan, K., Yan, X., Yu, X., Xu, C.S.: Real-time goalmouth detection in MPEG soccer video. In: Proc. of ACM MM 2003, Berkeley, USA, pp. 311–314 (2003)
17. Wan, K., Yan, X., Yu, X., Xu, C.S.: Robust goalmouth detection for virtual content insertion. In: Proc. of ACM MM 2003, Berkeley, USA, pp. 468–469 (2003)
18. Wang, J.R., Paramesh, N.: A scheme for archiving and browsing tennis video clips. In: Proc. of IEEE Pacific-Rim Conf. on Multimedia - PCM 2003, Singapore (2003)
19. Wang, J.R., Parameswaran, N.: Survey of Sports Video Analysis: Research Issues and Applications. In: Conferences in Research and Practice in Information Technology (VIP 2003), Sidney, vol. 36, pp. 87–90 (2004)
20. Yow, D., Yeo, B.L., Yeung, M., Liu, B.: Analysis and presentation of soccer highlights from digital video. In: Li, S., Teoh, E.-K., Mital, D., Wang, H. (eds.) ACCV 1995. LNCS, vol. 1035. Springer, Heidelberg (1996)
21. Yu, X., Xu, C.S., Leong, H.W., Tian, Q., Tang, Q., Wan, K.W.: Trajectory-based ball detection and tracking with applications to semantic analysis of broadcast soccer video. In: Proc. of ACM MM 2003, Berkeley, USA, pp. 11–20 (2003)
22. `http://www.hawkeyeinnovations.co.uk/`
23. `http://www.questec.com/`

24. Röfer, T., et al.: German Team 2005 Technical Report, RoboCup 2005, Four-legged league (February 2006), http://www.germanteam.org/GT2005.pdf
25. Quinlan, M.J., et al.: The 2005 NUbots Team Report, RoboCup 2005, Four-legged league (February 2006), http://www.robots.newcastle.edu.au/publications/NUbotFinalReport2005.pdf
26. Loncomilla, P., Ruiz-del-Solar, J.: Gaze Direction Determination of Opponents and Teammates in Robot Soccer. In: Bredenfeld, A., Jacoff, A., Noda, I., Takahashi, Y. (eds.) RoboCup 2005. LNCS, vol. 4020, pp. 230–242. Springer, Heidelberg (2006)
27. Ruiz-del-Solar, J., Loncomilla, P., Vallejos, P.: An automated refereeing and analysis tool for the Four-Legged League. In: Lakemeyer, G., Sklar, E., Sorrenti, D.G., Takahashi, T. (eds.) RoboCup 2006: Robot Soccer World Cup X. LNCS, vol. 4434, pp. 206–218. Springer, Heidelberg (2007)
28. Arenas, M., Ruiz-del-Solar, J., Verschae, R.: Detection of Aibo and Humanoid Robots using Cascades of Boosted Classifiers. In: Visser, U., Ribeiro, F., Ohashi, T., Dellaert, F. (eds.) RoboCup 2007: Robot Soccer World Cup XI. LNCS, vol. 5001, pp. 449–456. Springer, Heidelberg (2008)
29. Ruiz-del-Solar, J.: Personal Robots as Ubiquitous-Multimedial-Mobile Web Interfaces. In: 5th Latin American Web Congress LA-WEB 2007, Santiago, Chile, October 31 – November 2, pp. 120–127 (2007)
30. Comaniciu, D., Ramesh, V., Meer, P.: Kernel-Based Object Tracking. IEEE Trans. on Pattern Anal. Machine Intell. 25(5), 564–575 (2003)
31. RoboCup 2008 Humanoid official rules (January 2008), http://mail.fibo.kmutt.ac.th/robocup2008/HumanoidLeagueRules2008draft-2008-01-20.pdf
32. Peng, N.S., Yang, J., Liu, Z.: Mean shift blob tracking with kernel histogram filtering and hypothesis testing. Pattern Recognition Letters 26, 605–614 (2005)
33. Lowe, D.G.: Distinctive Image Features from Scale-Invariant Keypoints. Int. Journal of Computer Vision 60(2), 91–110 (2004)
34. Bender robot official website, http://bender.li2.uchile.cl/
35. CSLU toolkit Official Website, http://cslu.cse.ogi.edu/toolkit/