Design Issues for a Goalkeeper Robot

Riccardo Cassinis, Alessandro Rizzi
Dept. of Electronics for Automation - University of Brescia
Via Branze 38, I-25123 Brescia - Italy
Phone +39-030-3715.453
E-mail: {riccardo.cassinis, alessandro.rizzi}@unibs.it

Abstract

This paper presents Saracinescu, the goalkeeper robot of the Italian team that was used at the Robocup 98 Paris championship. The machine features an original omni-directional vision system whose performance, enhanced by a simple but effective movement strategy, proved to be very smart and led to good results during the tournament. The paper describes the vision algorithms in detail, and discusses some issues that are still being developed and/or refined. An overview of the other components of the machine (mechanical structure and ball-kicking mechanism, computing architecture, auxiliary software routines for initial positioning, etc.) is also included.

1 Introduction

The Robocup championship offers a simple and well-structured environment, suitable for testing some innovative robot features like the visual guidance system presented in this paper. One of the simplifications that this environment introduces with respect to the real world is the small number of colors used in the playground and the rigid coding of their meaning. However, even if each game component has a unique color, some problems for color matching still remain, due to illumination changes, shadows, reflections, etc.

The robot we present is the goalkeeper of the Italian team (ART), which exhibited a very good performance during the Robocup 98 Paris championship. Its main characteristics are an omni-directional vision system and a simple but effective reactive strategy.

The visual guidance system is based on color information grabbed with an omni-directional, quasi-spherical device. The intrinsic geometric complexity of the image is simplified by the use of color: neither shape nor other geometric features are taken into account. Only the colors and the relative position of the objects surrounding the robot influence its movement.

Even if the idea of using omni-directional visual devices had already been used in previous Robocup events, the presented one allows measuring not only the direction, but also the distance of relevant objects. The underlying idea has since then been adopted by several other researchers involved in the Robocup competition, both for goalies and for other players.

In Par. 2, the overall structure of the robot is presented, while the details of the vision subsystem are described in Par. 3, and the operation strategy is introduced in Par. 4.

2 The structure of the robot

The robot (Figure 1) is based on a modified version of the widely used, commercial RWI Pioneer 1 platform. An on-board PC, with the appropriate power supply, was added to provide the necessary computing power. The PC (an Intel Pentium II) runs LINUX operating system, and is equipped with all the necessary peripherals, that include an Intel Video Recorder frame grabber used to acquire camera images, and a Wavelan wireless networking interface.

Given the task the goalie has to accomplish, the only possibility offered by the mechanical structure was to use the robot sideways, in order to make it able to quickly reach any point of the goal area. The original castor wheel that supports the weight of the robot was replaced with a spherical device in order to eliminate lateral skids when the robot reverses its movement.

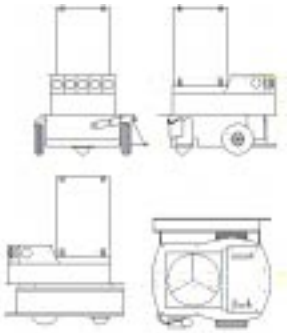

Fig. 1. Saracinescu structure design.

A mechanical ball-kicking device has been mounted on the left side of the robot, that always faces the play-

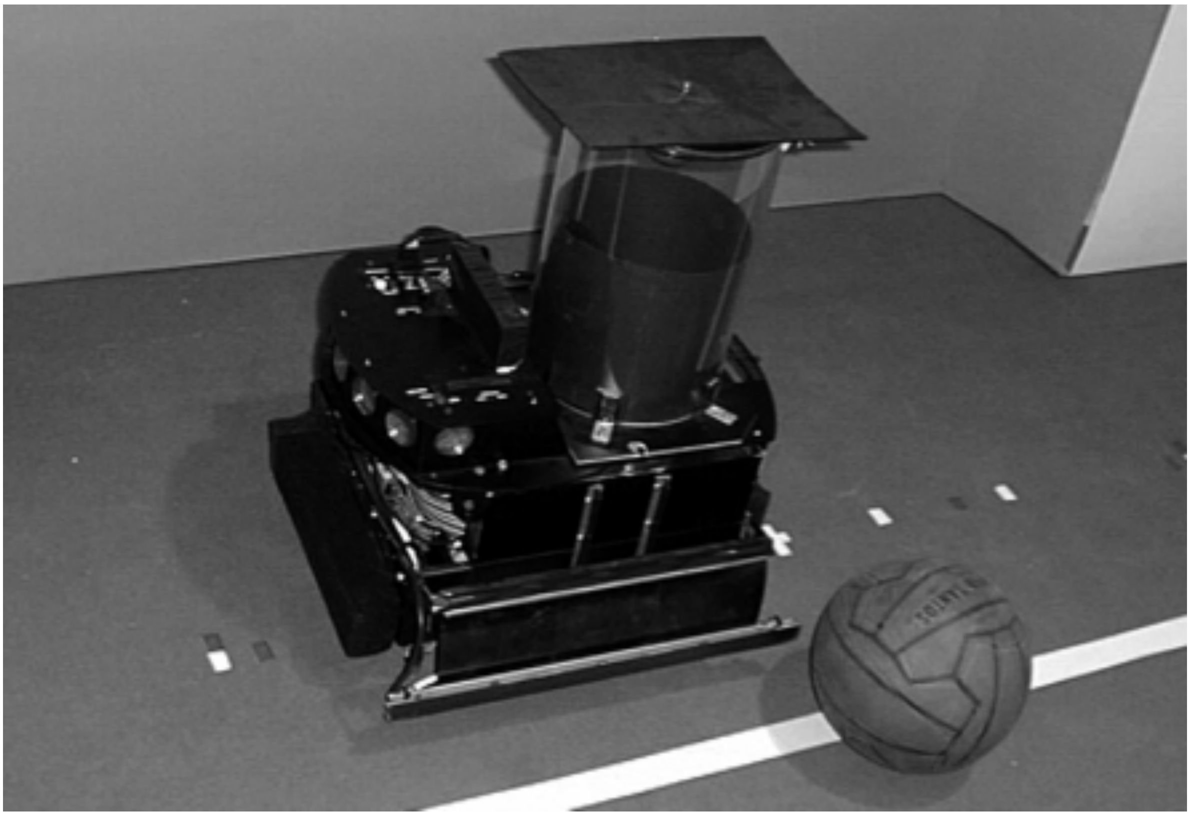

Fig. 2. Saracinescu robot.

ground (Figure 3). When the ball touches the bar placed at the lower edge of the kicker, it activates a mechanical switch that in turn triggers a kick and reload mechanism. Kicking power is provided by a steel spring, while reloading is accomplished by an electric motor. After each kick, it takes about two seconds to reload the mechanism. The kicking reflex takes place locally, i.e. with no computer intervention: the main computer can, however, disable and re-enable the device if required.

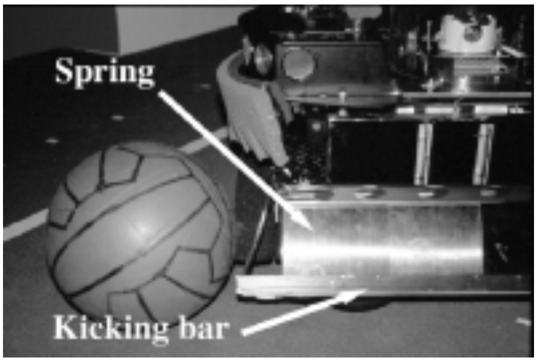

Fig. 3. The kicking mechanism.

Another significant change was the addition of a large Plexiglas tube, clearly visible in Figure 2, that was mounted on the upper part of the robot and that supports the mirror of the vision system, as will be described in the following paragraph.

As far as the other sensors are concerned, four active infrared proximity sensors were added under the robot body. The idea was to use them to detect the white lines that mark the goal and the penalty area, in order to provide the robot with additional information about its position. However, the omni-directional vision system was found accurate and reliable enough, and the proximity sensors, although interfaced, were never used.

The standard sonars that come with Pioneer robots were also not used, because they cannot be relied upon when they work in an environment cluttered with other robots using their own (and probably similar) sonars.

3. The omni-directional visual device

Several kinds of omni-directional visual sensors, with different geometrical characteristics [Yagi *et al.*, 1995; Yagi *et al.*, 1994; Yamazawa *et al.*, 1993] and different optical pre-processing capabilities [Cassinis *et al.*, 1996] have been investigated so far. They have been used for various robot navigation and self-localization tasks.

The omni-directional device developed for Saracinescu uses a mirror with a spherical sector shape and an optical-grade reflecting surface, that allows a clear vision of what is happening around the robot. The spherical shape of the mirror allows the perception of a larger amount of detail in the area surrounding the robot and only rough visual information of the area far from the robot. The mirror axis is vertical, and the device (actually, a 20 cm diameter stainless steel pan lid was used) is supported by a clear Plexiglas tube, that also houses the camera. The idea is to mimic the behavior of a real goalkeeper that does not care much about what is going on in the opposite half of the field, but pays great attention when the ball comes close to him.

Figure 4, besides giving an idea of the structure of the device, shows its most interesting feature: the shape of the mirror allows measuring not only the bearing of the objects with respect to the robot, but also their apparent horizontal elevation. Since all objects in Robocup lay on the ground and have known dimensions, the system can compute object distances as well.

An upward pointing CCD color camera grabs the images reflected by the mirror. Its signals are processed in order to extract information about the goalposts and the ball. This information is used to keep the robot between the ball and the goal.

Figure 5 shows an actual image grabbed from the camera. Grey scale rendering makes it hard to distinguish details, but it is quite easy to recognize the ball (originally red), the field walls and lines (white) and the goal (black, in the lower part of the image). The figure shows the results of processing superimposed to the original image: the recognized portion of the ball is drawn in blue, while the goal (originally yellow) is drawn in black. Careful analysis of the picture shows that the entire playground is shown, including the opposite goal. Far objects are poorly detailed but, as it was said, this is not important for the goalie.

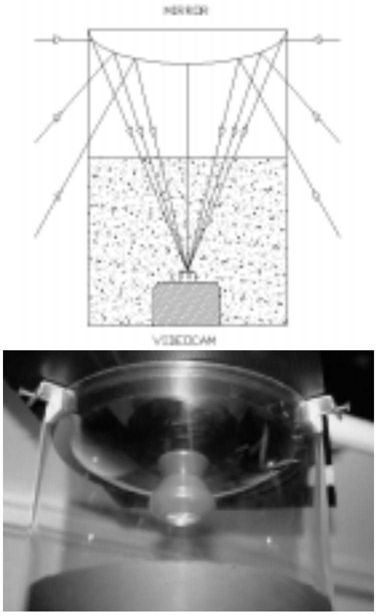

Fig. 4. The omni-directional vision system and its mirror.

Geometrical calibration of the camera-mirror system can be accomplished in several ways, using mechanical or optical devices. The procedure that was used is as follows: once the camera has been lined up with the transparent tube using a cardboard disc with a mark in its center, the mirror position is adjusted using the cross that marks its center, and that is well visible in Figures 4 and 5. This procedure is far from perfect, but yields sufficient precision for this application.

Measuring the Euclidean distance of objects from the center of the image allows estimating their distance from the robot, if their height above the ground level is known. Since the exact shape of the mirror is not known, it would have been hard to determine the equation that yields the actual distance as a function of the distance measured on the image. Instead, a look-up table was experimentally built, that lists correspondences between 50 and 230 cm (the estimated useful range), in 10 cm increments. The system uses linear interpolation for estimating intermediate values.

3.1 Color segmentation and object recognition

As it was said in the introduction, Robocup rules assign unique colors to all the objects in the playground. Therefore, no shape information was used for object recognition.

Two constraints influence the color segmentation process: the high computing speed required to effectively track the ball and the heavily varying illumination conditions. Preliminary tests of the robot conducted on various training playgrounds have shown that apparent colors of the objects vary from field to field, due to the different light sources, materials used for building the field, etc. Additionally, our system has to deal with its very low-cost surveillance camera, whose color rendering and constancy are light-years far from perfection. Shadows produced by objects make the task even more tough.

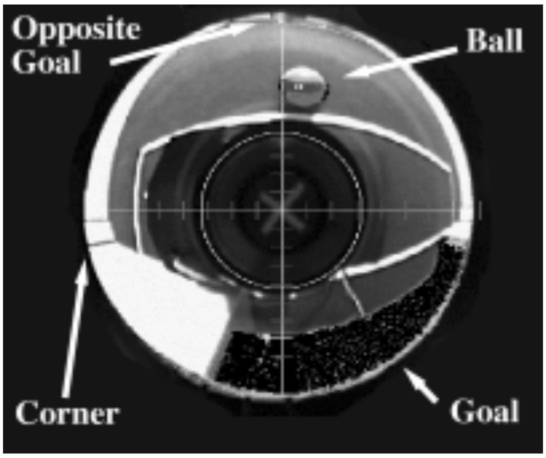

Fig. 5. A segmented omni-directional image.

To solve the problem of widely varying light conditions, a pre-match calibration phase has been introduced. During this phase, usually performed once before each match half, a supervisor manually selects a small area of the image that belongs completely to the ball. The mean chromatic value among all the selected pixels is computed. This triplet is taken as the center of a parallelepiped in the RGB space that will be used to cluster pixels belonging to the ball. Its sides lengths are initially set equal to the relative chromatic variance of the selected pixels. This leads to a quite robust automatic clustering, but some problems still occur in presence of shadows and reflections.

Thus, a reinforcement of the color calibration phase has been introduced. In this second phase, the supervisor selects a larger area of the image, that completely encloses the ball. The three RGB thresholds are then incremented, thus enlarging the color clustering subspace, until a pixel that surely does not belong to the ball is selected.

After the calibration on the ball color, the same procedure is used to determine the color of the goal that can be alternatively yellow or blue. The whole procedure could be automated, but due to the strict deadline of the tournament, this has been deferred. Actually, it only takes about one minute to manually train the system.

3.2 Ball and goalposts recognition

During normal operation, the goal of the vision system is to recognize the ball and the goalposts, and to measure their position with respect to the robot. The speed requirements call for processing at least 10 frames per second, and some optimization had to be done to reach this goal with the available computing hardware.

The central part of the image does not contain any useful information, and is discarded. Only the external circular part of the image is considered for searching the ball and the goalposts. With the mounted low-cost commercial camera, five bits per pixel for each chromatic channel have been used. A higher resolution would have been completely useless.

The color segmentation is performed using the previously described *region-growing* algorithm, which easily detects the ball. The goalposts are then detected selecting the boundary pixels between the goal color and the white field walls. Due to the geometry of the image, no confusion between the left and the right goalpost can arise.

The ball is localized using the apparent center of gravity of the cluster of red pixels. To avoid localizing errors, a threshold on the minimum number of clustered pixels is used. The posts can be localized using either the middle or the lower point of the detected boundary line. In the first case perspective errors due to the changing distance of the robot and the goal gate are introduced. In the second case, the shadow of the robot may cover the lower post pixels introducing errors as well. The tests and the played matches proved that both methods work correctly in most practical cases. Regardless of the chosen method, what is really important for the robot movement, is not the goalposts distance estimate, but their relative angles.

The computing speed requirements, and the restrictions of the Pioneer software system, suggested that the ball and the goal recognition should be treated as two separate activities. Moreover, since the ball moves much faster than the robot, two ball searches are performed per each goal recognition.

4 Strategy and robot movement

All the strategies described in the following paragraphs have been implemented in C using the *activities* functionality of the Saphira programming environment which contains all the primitives for the robot movement. It was decided to avoid using Saphira *behaviors* because, even if they can help solving some of the problems, they add an unacceptable computing burden, making the robot too slow for any practical use.

Using Saphira activities and *micro-tasks* concept, different processes can be executed sequentially and cyclically within predefined time slices. In this way, various programs can run together with a simulated parallelism.

The basic strategy obviously is to keep the robot between the ball and the goal gate at all times. In order to do that, Saracinescu lays its right side toward the goal gate and moves back and forth like in Figure 6. In this position, the kicking mechanism faces the playground.

Intercepting the ball only requires straight movements parallel to the goal: however, wheel skidding, encoder tolerance and collisions with other robots can result in involuntary robot rotations. To correct this problem, at regular time intervals, a process to control and correct the robot horizontal parallelism is executed.

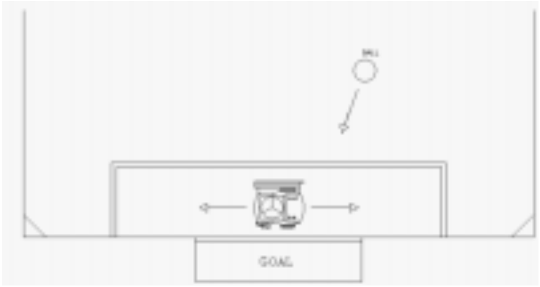

Fig. 6. Saracinescu basic strategy.

Furthermore, since a constant angle between two reference points results in an arc, measuring the bearing of the goalposts does not allow determining the actual distance of the robot from the goal. Therefore, another task is started at regular time intervals. It controls the estimated distance of the goalposts and moves the robot towards or away from the goal if required.

This activity is also used at the beginning of the game, when the robot is placed anywhere on the field and must locate and reach its goal.

Summarizing, there are four main logical activities running concurrently during normal robot operation:

- grab the omni-directional image and extract distance and angle of the ball and the goalposts,
- keep the robot between the ball and the goal,
- keep the robot parallel to the goal gate,
- keep the distance between the robot and the goal constant.

All the geometrically derivable data are directly extracted from the image whenever possible, introducing data redundancy. If, due to visual noise (occlusion, shadows, etc.) some data are missing, estimates from the data series support the robot for a limited time interval. After this period, if data are still missing, the robot enters a stall state until a sufficient amount of data is available again.

In order to obtain a not too nervous robot behavior and thus not to overstress the motors, a low-pass filter that averages data in time has been applied. It can be seen as an inertia increment that allows Saracinescu to keep a steady position without high frequency oscillation.

4.1 The goalkeeping strategy

The goalkeeping strategy is to maintain the line that connects the robot to the ball coincident with the bisecting line of the angle formed by the goalposts and the robot (Figure 7). In order to accomplish this strategy, this activity moves the robot only along a line parallel to the goal (Figure 6).

The bisecting line has been chosen instead of the median because it better protects the goal gate. The median would put the robot too close to the goal center. On the contrary, the bisecting line better protects the goalpost surroundings (Figure 7), stopping also most of the bouncing kicks.

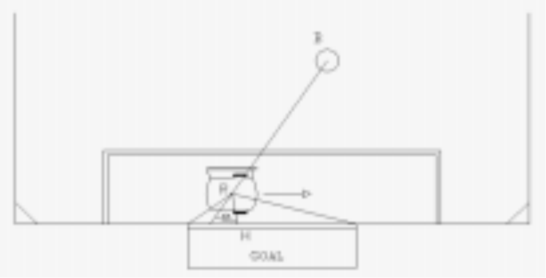

Fig. 7. The goalkeeping strategy.

The algorithm includes provisions for preventing the robot from moving out of the goal area. Whenever the robot gets close to a goalpost, the main positioning algorithm is excluded and the robot starts to decrease velocity, in order to stop at the goalpost point. Overtaking this point, in fact, is useless.

The parallelism keeping activity controls the angle between the robot heading and the goal line. The smallest angle the Pioneer can rotate is 5 degrees, thus angles below this value are not considered. Moreover, the Pioneer firmware has its own accidental rotation automatic corrector, so it is important to insert a delay before triggering this activity.

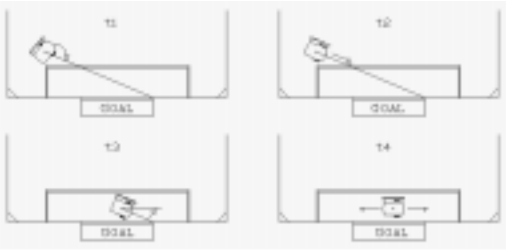

Fig. 8. The repositioning algorithm.

The goal distance control activity controls the robot distance from the goal. If the distance exceeds a threshold value, all the activities are stopped and a repositioning algorithm is executed. This algorithm drives the robot towards its default starting position as shown in Figure 8.

5 Conclusions and perspectives

The performance of Saracinescu, that has been the official goalkeeper of the Italian Azzurra Robot Team at the Robocup 1998 championship in Paris, has gone beyond the most optimistic expectances. Its reactive behavior lacks any reasoning and forecast capability, but the machine was perfectly apt for the simple task it had to accomplish. The robustness of the vision system was however a determining factor for its success. It localized the ball during all the matches with good precision and regardless of the illumination, of the shadows and of reflections in the playground. This suggests that all the robots in a team, and not only the goalkeeper, should use a similar system for visual data acquisition.

It can be expected, however, that the performance of other robots will increase in future championships. Several improvements should then be made to the structure of our goalie. Besides the mechanical requirements, that suggest to completely rebuilding the machine in order to have a much lighter and faster robot, it would be desirable to fully automate the color calibration procedure, using an automatic ball and goalpost detection method. Object recognition should be performed using predictive algorithms, in order to speed up the process, first looking for objects in the places where they are expected to be. Recognition of other robots should be introduced, in order to allow reasoning about where the ball will be kicked by an opponent robot, and how it should be usefully passed to a teammate. Finally, it would be useful to implement some sort of communication with the rest of the team, in order to better coordinate the defense action.

Acknowledgments

The authors wish to acknowledge the *Consorzio Padova Ricerche* and its director, Prof. Giorgio Clemente, for having provided partial funding for the construction of the robot described in this paper and for having sponsored the participation of the robot at 1998 Paris Robocup Championship.

References

[Cassinis *et al.*, 1996] R. Cassinis, D. Grana, A. Rizzi. Using Colour Information in an Omnidirectional Perception System for Autonomous Robot Localization. In *Proceedings of EUROBOT96 First Euromicro Workshop on Advanced Mobile Robots '96,* Kaiserslautern (Germany), 1996.

[Yagi *et al.*, 1994] Y. Yagi, H. Okumura, M. Yashida. Multiple Visual Sensing System for Mobile Robot. In *Proceedings of the IEEE Int. Conference on Robotics and Automation,* 1994, Vol 2, pp. 1679-1684.

[Yagi *et al.*, 1995] Y. Yagi, Y. Nishizawa, M. Yashida. Map based Navigation for a Mobile Robot with Omnidirectional Image Sensor COPIS. *IEEE Trans. Robotics and Automation,* Vol.11, No 5, 1995.

[Yamazawa *et al.*, 1993] K. Yamazawa, Y. Yagi, M. Yachida. Omnidirectional Imaging with Hyperboloidal Projection. In *Proceedings of the IEEE/RSJ IROS'93,* Vol. 2, pp. 1029-1034, 1993.