Advances in Artificial Intelligence - IBERAMIA 2004, pp. 535-544. Lemaître C. et al. (eds). LNAI 3315, Springer-Verlag Berlin Heidelberg (2004). ISBN 3-540-23806-9.

Development of Local Perception-Based Behaviors for a Robotic Soccer Player

Antonio Salim*, Olac Fuentes, and Angélica Muñoz

National Institute of Astrophysics, Optics and Electronics, Luis Enrique Erro # 1,
Santa María Tonantzintla, Puebla, 72840, México
{asalimm,fuentes,munoz}@inaoep.mx

Abstract. This paper describes the development of local vision-based behaviors for the robotic soccer domain. The behaviors, which include *finding ball, approaching ball, finding goal, approaching goal, shooting* and *avoiding*, have been designed and implemented using a hierarchical control system. The *avoiding* behavior was learned using the C4.5 rule induction algorithm, the rest of the behaviors were programmed by hand. The object detection system is able to detect the objects of interest at a frame rate of 17 images per second. We compare three pixel classification techniques; one technique is based on color thresholds, another is based on logical AND operations and the last one is based on the artificial life paradigm. Experimental results obtained with a Pioneer 2-DX robot equipped with a single camera, playing on an enclosed soccer field with forward role indicate that the robot operates successfully, scoring goals in 90% of the trials.

1 Introduction

Robotic soccer is a common task for artificial intelligence and robotics research [1]; this task permits the evaluation of various theories, the design of algorithms and agent architectures. This paper focuses on the design and evaluation of perceptual and behavioral control methods for the robotic soccer domain; these methods are based on local perception, because it permits designers to program robust and reliable robotic soccer players that are able to cope with highly dynamic environments such as RoboCup environments.

Vision is the primary sense used by robots in RoboCup. We used a local vision approach with an off-board computer. In this approach, the robot is equipped with a camera and an off-board image processing system determines the commands for the robot. We used this approach because of the advantages that it offers, which include lower power consumption, faster processing and the fact

* The first author is supported by CONACyT, under grant number 171617.

that inexpensive desktop computers can be used instead of specialized vision processing boards. We compare three strategies for pixel classification. One strategy is based on color thresholds [8], another is based on the algorithm of Bruce et al. [6] and the last one is based on the artificial life paradigm.

Behaviors were designed and implemented using a hierarchical control system with a memory module for a reactive robotic soccer player [5]. The behaviors, which include *finding ball, approaching ball, finding goal, approaching goal,* and *shooting*, were programmed by hand. The *avoiding* behavior was learned via direct interaction with the environment with the help of a human operator using the C4.5 rule induction algorithm [9].

The paper is organized as follows. Section 2 reviews related work. Section 3 describes the methodological approach used in the design of our robotic soccer player. Section 4 sumarizes the experimental results obtained. Finally, Section 5 discusses conclusions and perspectives.

2 Related Work

We survey a number of works in the field of vision and control for robotic soccer.

2.1 Vision

The cognachrome vision system©, manufactured by Newton Research Labs, is a commercial hardware-based vision system used by several robot soccer teams [13]. Since it is hardware-based, it is faster than software running on a general-purpose processor. Its disadvantages are its high cost and the fact that it only recognizes three different colors.

A number of past RoboCup teams have used alternative color spaces such as HSB or HSV for color discrimination proposed by Asada [2], since these separate color from brightness reducing sensitivity to light variations.

Several RoboCup soccer teams have adopted the use of omnidirectional vision generated by the use of a convex mirror [3]. This type of vision has the advantage of providing a panoramic view of the field, sacrificing image resolution. Moreover, the profiles of the mirrors are designed for a specific task.

The fast and cheap color image segmentation for interactive robots employs region segmentation by color classes [6]. This system has the advantage of being able to classify more than 32 colors using only two logical AND operations and it uses alternative color spaces.

For our vision system, we used the pixel classification technique proposed by Bruce [6] and a variant of the color spaces proposed by Asada [2] (see Section 3.2).

2.2 Control

Takahashi et al. [12] used multi-layered reinforcement learning, decomposing a large state space at the bottom level into several subspaces and merging those subspaces at the higher level. Each module has its own goal state, and it learns to reach the goal maximizing the sum of discounted reward received over time.

Steinbauer et al. [11] used an abstract layer within their control architecture to provide the integration of domain knowledge such as rules, long term planning and strategic decisions. The origin of action planning was a knowledge base that contained explicit domain knowledge used by a planning module to find a sequence of actions that achieves a given goal.

Bonarini et al. [4] developed a behavior management system for fuzzy behavior coordination. Goal-specific strategies are reached by means of conflict resolution among multiple objectives. Behaviors can obtain control over the robot according to fuzzy activation conditions and motivations that reflect the robot's goals and situation.

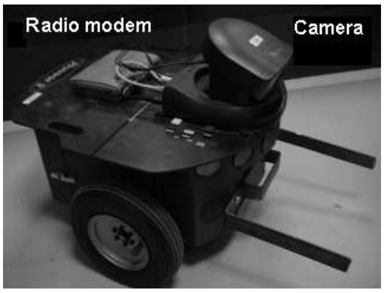
Gómez et al. [7] used an architecture called dynamic schema hierarchies. In this architecture, the control and the perception are distributed on a schema collection structured in a hierarchy. Perceptual schemas produce information that can be read by motor schemas to generate their outputs.

We used a behavior-based control system or subsumption architecture with a memory module in order to control our robotic soccer player (see Section 3.3).

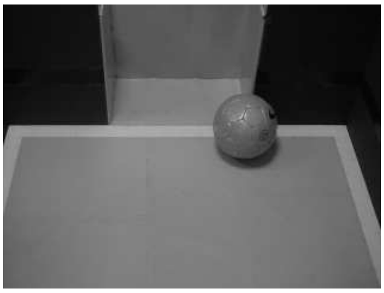
3 The System

3.1 Hardware and Settings

The robot used in this research is a Pioneer 2-DX mobile robot made by ActivMedia©, equipped with a Pioneer PTZ camera, a manually-adapted fixed gripper and a radio modem. The dimensions of the robot are 44 cm long, 38 cm wide and 34 cm tall, including the video-camera. The robot is remotely controlled by a AMD Athlon 1900 computer with 512 MB of RAM. Figure 1(a) shows a picture of our robotic soccer player.



(a)



(b)

Fig. 1. The robotic soccer player (a). The soccer playing field (b)

The environment for the robot is an enclosed playing field with a size of 180 cm in length and 120 cm in width. There was only one goal, painted cyan, centered in one end of the field with a size of 60 cm wide and 50 cm tall. The walls were marked with an auxiliary purple line whose height is 20 cm from the floor. Figure 1(b) shows a picture of the playing field.

3.2 Vision

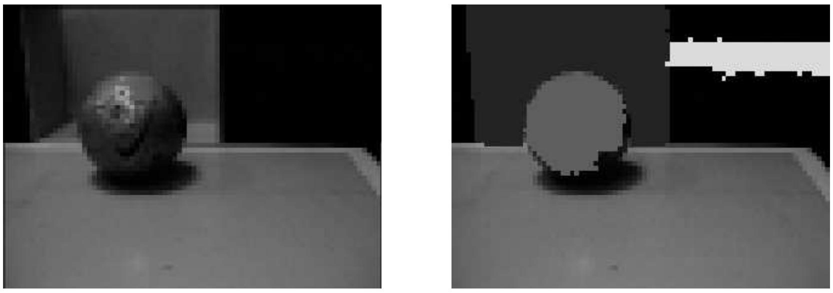
A robust, fast and fault tolerant vision system is fundamental for the robot, since it is the only source of information about the state of the environment. Since all objects of interest in the environment are colored, we believe that vision is the most appropriate sensor for a robot that has to play soccer. We present below the object detection system used by the robot and a strategy for pixel classification based on the artificial life paradigm.

Object Detection. The vision system processes images captured by the robot's camera and reports the locations of various objects of interest relative to the robot's current location. The objects of interest are the orange ball, the cyan goal and the auxiliary purple line on the field's wall. The steps of our object detection method are:

1. *Image capture:* Images are captured in RGB in a 160×120 resolution.
2. *Image resizing:* The images are resized to 80×60 pixels.
3. *Color space transformation:* The RGB images are transformed into the HUV color space, for reducing sensitivity to light variations.
4. *Pixel classification:* Each pixel is classified by predetermined color thresholds in RGB and HUV color spaces. There are three color classes: the colors of the ball, the goal, and the auxiliary line. The pixel classification is based on [6], in order to use only two logical AND operations for each color space.
5. *Region segmentation:* Pixels of each color class are grouped together into connected regions.
6. *Object filtering:* False positives are filtered out via region size.

Figure 2(a) shows an image captured by the frame grabber and Figure 2(b) shows the robot's perception.

Artificial Life Approach for Pixel Classification. In order to reduce the time invested in pixel classification, the most expensive step in object detection, we tested an artificial life-based method. Ideas of distributed computing were taken from Reynolds's boids [10], where a group of agents moves as a flock of birds or a school of fish. For this strategy, we used 2500 agents, each having an internal state to indicate whether it is over an object of interest or not. Agents were able to detect three color classes: the colors of the ball, the goal and the auxiliary line in the walls. Agents were serialized by an agent manager which assigned movement turns and prevented collisions between agents. However, the



(a) (b)

Fig. 2. Image captured by the camera (a). The robot's perception (b)

recognition task is distributed among agents. The agents can move in their world, which is the image perceived by the camera. Only one agent can be located over each pixel. Agents can sense the color intensity values in the image in order to perform pixel classification. The locomotion of an agent consists of moving pixel by pixel via its actuators. Figure 3 shows a snapshot of the pixel classification method based on Artificial life.

Fig. 3. Artificial life-based pixel classification

3.3 Control

Behaviors were designed and implemented using a subsumption architecture [5] because this architecture offers the necessary reactivity for dynamic environments. We incorporated a new element to this architecture, a memory module. This module acts as a short-term memory that enables the robot to remember past events that can be useful for future decisions. The memory module affects directly the behaviors programmed into the robot.

The *avoiding* behavior is a horizontal behavior in the architecture that overwrites the output of the rest of the behaviors in our vertical subsumption architecture. The architecture was implemented using four threads in C++, one for the vertical behaviors module, one for the memory module, one for controlling the robot movements and one for the horizontal behavior to avoid collisions with the walls. In this architecture, each behavior has its own perceptual input, which is responsible of sensing the objects of interest. Each behavior writes its movement commands to shared memory to be executed. The architecture used for the robot's control system is shown in Figure 4.

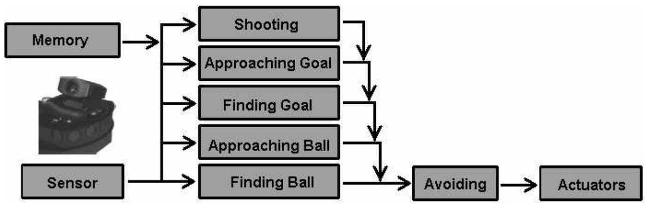


Fig. 4. The architecture of the system

3.4 Description of Modules and Behaviors

1. *Memory:* This is an essential module for the achievement of the robot's global behavior. Memory, like behaviors, has its own perceptual input to sense the ball and the goal. The function of this memory is to remember the last direction in which the ball or the goal were perceived with respect to the point of view of the robot. The memory module affects directly the other behaviors because it writes the directions of the ball and the goal on a shared memory used in the behaviors's execution. There are six possible directions that the memory has to remember: ball to the left, ball to the right, centered ball, goal to the left, goal to the right and centered goal.
2. *Finding ball:* The robot executes a turn around its rotational axis until the ball is perceived. The robot turns in the direction in which the ball was last perceived. If this information was not registered then the robot executes a random turn towards the left or right.
3. *Approaching ball:* The robot centers and approaches the ball until the ball is at an approximate distance of 1 cm.
4. *Finding goal:* The robot executes a turn around its rotational axis until the goal is perceived. The robot turns in the direction in which the goal was last perceived. If this information was not registered then the robot executes a random turn towards the left or right.
5. *Approaching goal:* The robot executes a turn in the direction of the center of the goal until the goal is centered with respect to the point of view of the robot.

6. *Shooting:* The robot makes an abrupt increase of its velocity to shot the ball towards the goal. There are two possible kind of shots, a short shot when the robot is close to the goal (a distance equal or less than 65 cm) and a long shot, when the robot is far from the goal (more than 65 cm).
7. *Avoiding:* The robot avoids crashing against the walls that surround the soccer field. Determining manually the necessary conditions in which the robot collides with the wall is difficult because the wall can be perceived in many forms, therefore we used the machine learning algorithm C4.5 [9] to learn whether a collision must be avoided or not.

4 Experimental Results

4.1 Pixel Classification Results

We present the results obtained by three implementations of pixel classification. The first implementation was based on color thresholds [8], the second implementation was based on the algorithm proposed by Bruce et al. for pixel classification [6], and finally, the third implementation was based on the artificial life paradigm.

Table 1. Pixel classification results

Method	Images per second	Processing average time
Color thresholds	12 images	0.0874 sec.
Bruce-based method	18 images	0.0553 sec.
Artificial life-based method	14 images	0.0707 sec.

Results of pixel classification are shown in Table 1. As this table indicates, the worst strategy for pixel classification task was based on color thresholds [8]. The best strategy for this task was based on the algorithm proposed by Bruce et al. [6], this strategy was implemented as a step in the object detection system for the robotic soccer player. We expected a better performance from the pixel classification method based on artificial life, because this method needs to examine only 2500 pixels, corresponding to the total number of agents, instead of the total number of pixels in the image (8600 pixels). However, in this strategy each of the agents spends time calculating its next movement, producing a general medium performance.

4.2 Avoiding Behavior Results

For the *avoiding* behavior, we collected a training set of 446 instances of collisions. There were 153 positive samples where there was a collision and 293 negative samples where there was not collision. The elements of the input vector were *roundness, compactness, convexity, orientation, contour length, mean* *length of runs, line index of lower right corner point, column index of lower right corner point, row of the largest inner circle* and *column of the largest inner circle* of the ball's region detected by the object detection system. The experiments were validated using 10-fold cross-validation. We tested 5 machine learning algorithms for the classification task; the results obtained are summarized in Table 2. As this table shows, the C4.5 algorithm obtained the best percentage of correctly classified instances for the collision avoidance task. The rules generated by C4.5 algorithm were implemented in our *avoiding* behavior.

Table 2. Percentage of correctly classified instances by machine learning algorithm for the *avoiding* behavior

Machine learning algorithm	% of correctly classified instances
Support Vector Machines	91.25 ± 0.0603 %
Artificial Neural Networks	90.40 ± 0.0849 %
C4.5	92.20 ± 0.0638 %
Naive Bayes	87.62 ± 0.0683 %
Conjuntive Rules	90.68 ± 0.0222 %

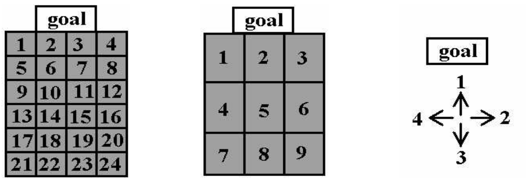
4.3 Global Performance

Our robotic soccer player has a forward role, thus its main task is to score goals in a minimum amount of time. In order to test the global performance of our robotic soccer player, we designed a set of experiments. The experiments were performed on the soccer field shown in Figure 1(b). The robot position, robot orientation and ball position were selected 20 times randomly as follows:

1. For selecting the robot's position, the field was divided into 24 cells of equal size. Figure 5(a) shows the cells for the robot position.
2. For selecting the ball's position, the field was divided into 9 cells of equal size. Figure 5(b) shows the cells for the ball position.
3. For selecting the robot's orientation, there were 4 directions to the robot. The orientation where the goal is: 1) in front of the robot, 2) left to the robot, 3) back to the robot and 4) right to the robot. Figure 5(c) shows the possible orientations for the robot.

An experiment's configuration can be represented as a triplet, of the form *(ball position, robot position, robot orientation)*. The configuration for the 20 experiments performed were: *(24,7,1), (24,8,1), (21,2,2), (8,8,4), (18,7,3), (22,9,2), (24,4,4), (7,4,3)* , *(6,4,3), (8,2,2), (15,1,3), (21,4,1), (12,2,2), (11,9,1), (7,8,4), (20,9,1), (7,9,4), (11,9,4), (10,5,2)* and *(6,2,3)*. Table 3 summarizes the time spent in seconds by each behavior performed by the robot in the experiments. The total time spent by the robot in the experiments was 632 seconds.

The percentage of time used by behaviors in the experiments was 28% for *Finding Ball*, 32.27% for *Approaching Ball*, 14.24% for *Finding Goal*, 9.49% for



(a) (b) (c)

Fig. 5. Experiments's configuration. Robot position (a). Ball position (b). Robot orientation (c)

Approaching Goal and 16% for *Shooting*. The average time required by the robot to score a goal is 35.11 seconds.

The *avoiding* behavior was successful, the robot avoided 10 of 12 avoidance situations obtaining 83% success.

An useful functionality of the soccer player emerges from the interaction of three behaviors: *approaching ball*, *finding goal* and *avoiding*. This emergent behavior consist of *regaining the ball from the corner*. In the experiments, the robot was able to regain the ball from the corner four out of five times obtaining 80% success. In the 20 experiments executed, the robot was able to score 18 goals obtaining 90% success.

5 Conclusions

In this paper, we presented our research on the development of local perception-based behaviors for a Pioneer 2-DX robot equipped with a single camera.

The subsumption architecture used for the robot control gives the necessary reactivity to play soccer, also the memory that we incorporated enables the robot to base its decisions on past events.

The *avoidance* behavior was much easier to learn than to program by hand. Building the avoiding behavior using the C4.5 algorithm to learn to avoid collisions with the walls was successful.

Although the strategy for pixel classification based on artificial life did not improve the performance, it seems to be a promising strategy to create a completely distributed control system for a robotic soccer player. The main limitation of this approach is the current computational processing power needed to support a large number of agents with complex behaviors.

Using our object detection method we can detect the ball, goal and auxiliary line, at a frame rate of 17 frames per second.

Experimental results obtained with our robotic soccer player indicate that the robot operates successfully showing a high-level intelligent behavior and scoring goals in 90% of the trials.

Table 3. Time spent, in seconds, in each of the behaviors executed by the robot during 20 experiments. The symbol "-" indicates that a behavior in a given experiment was not executed. An experiment that contains only "-" symbols, is an unsuccessful experiment in which the robot got stuck against the walls

Experiment Number	*Finding Ball*	*Approaching Ball*	*Finding Goal*	*Approaching Goal*	*Shooting*	Duration
1	16	10	10	–	6	42
2	11	19	17	1	6	54
3	–	–	–	–	–	–
4	13	9	3	4	5	34
5	8	5	–	2	6	21
6	8	11	7	10	6	42
7	6	31	6	3	6	52
8	5	9	7	1	5	27
9	5	6	–	5	5	21
10	8	6	–	4	5	23
11	2	16	8	–	6	32
12	17	20	11	6	6	60
13	–	–	–	–	–	–
14	–	7	–	3	6	16
15	16	11	–	7	5	39
16	–	5	–	–	6	11
17	27	8	–	4	6	45
18	19	8	–	3	6	36
19	6	12	11	2	5	36
20	10	11	10	5	5	41
Totals	177	204	90	60	101	632 sec

In future work, we will use other machine learning techniques to help us develop behaviors such as *approaching ball*. The next step to reach in our research is to consider multi-robot coordination, an important issue in robot soccer.

References

1. Asada, M., Stone, P., Kitano, H., Werger, B., Kuniyoshi, Y., Drogoul, A., Duhaut, D., Veloso, M.: The RoboCup Physical Agent Challenge: Phase-I. Applied Artificial Intelligence. 12 (1998) 251–263.
2. Asada, M., Kitano, H.: RoboCup-98: Robot Soccer World Cup II. LNCS, Springer-Verlag. 1604 (1999).
3. Bonarini, A., Aliverti, P., Lucioni, M.: An omnidirectional vision sensor for fast tracking for mobile robots. Trans. on Inst. and Measurement. 49 (2000) 509–512.
4. Bonarini, A., Invernizzi, G., Labella, T. H., Matteucci, M.: An architecture to coordinate fuzzy behaviors to control an autonomous robot. Fuzzy Sets and Systems. 134 (2003) 101–115.
5. Brooks, R. A.: A Robust Layered Control System for a Mobile Robot. IEEE Journal of Robotics and Automation. RA-2 (1986) 14–23.

6. Bruce, J., Balch, T., Veloso, M.: Fast and inexpensive color image segmentation for interactive robots. In Proc. of the 2000 IEEE/RSJ IROS. (2000) 2061–2066.
7. Gómez, V. M., Cañas, J. M., San Martín, F., Mantellán, V.: Vision-based schemas for an autonomous robotic soccer player. In Proc. of IV WAF'2003. (2003) 109–120.
8. Pitas, I.: Digital Image Processing Algorithms and Applications. John Wiley & Sons Inc. (2000).
9. Quinlan, J. R.: C4.5: Programs for Machine Learning. Morgan Kaufmann, San Mateo, C.A. (1993).
10. Reynolds, C. W.: Flocks, Herds, and Schools: A Distributed Behavioral Model. Computer Graphics. 21 (1987) 25–34.
11. Steinbauer, G., Faschinger, M.: The Mostly Harmless RoboCup Middle Size Team. OGAI. 22 (2003) 8–13.
12. Takahashi, Y., Asada, M.: Vision-guided behavior acquisition of a mobile robot by multi-layered reinforcement learning. IEEE/RSJ IROS. 1 (2000) 395–402.
13. Werger, B. B., Funes, P., Schneider-Fontán, M., Sargent, R., Witty, C., Witty, T.: The Spirit of Bolivia: Complex behavior through minimal control. LNCS, Springer-Verlag. 1395 (1997) 348–356.