Visual Goal Detection for the RoboCup Standard Platform League

José M. Cañas, Domenec Puig, Eduardo Perdices and Tomás González

Abstract—**This paper presents a new fast and robust goal detection system for the Nao humanoid player at the RoboCup standard platform league. The proposed methodology is done totally based on Artificial Vision, without additional sensors. First, the goals are detected by means of color based segmentation and geometrical image processing methods from the 2D images provided by the front camera mounted in the head of the Nao robot. Then, once the goals have been recognized, the position of the robot with respect to the goal is obtained exploiting 3D geometric properties. The proposed system is validated with real images by emulating real RoboCup conditions.**

Index Terms—**RoboCup and soccer robots, Artificial Vision and Robotics, Nao humanoid, Goal Detection, Color Segmentation, 3D geometry.**

I. INTRODUCTION

THE RoboCup [1] is a scientific annual competition aimed to foster Robotics and Artificial Intelligence research. It offers soccer as a dynamic, competitive and cooperative benchmark for testing the robotics technology and pushing it forward. Its long term goal is to build a soccer team of robots able to beat the human world champion team by 2050. Maybe this is the equivalent to the long tearm milestone of AI community with artificial chess players, and Deep Blue defeated Gary Kasparov in 1997. The current state of the robotics technology is far from such ambitious goal, but progress has been made since the first RoboCup was celebrated in 1997.

In the last years several public challenges and competitions have arisen around robotics. For instance DARPA Grand Challenge and Urban Challenge have contributed to foster the research in robotics, providing proofs of concept about the feasibility of autonomous robot on real transportation missions.

RoboCup has worldwide scope and in the last years has included new categories beyond soccer: Junior, Rescue and Home. The last ones trying to reduce the gap between the contest and real applications. Several new leagues have also appeared around the soccer category, depending on the robot size and shape: small size, middle size, humanoid and standard

José M. Cañas and Eduardo Perdices are with the Rey Juan Carlos University. E-mail: jmplaza@gsyc.urjc.es

Domenec Puig and Tomás González are with the Rovira i Virgili University. E-mail: domenec.puig@urv.cat

[1] www.robocup.org

platform league (SPL). Maybe the most appealing one is the SPL as the hardware is exactly the same for all participants. The behavior quality and performance differences lie completely in the software. In addition, the code of the robots must be publicly described, so the knowledge sharing pushes the overall quality. Until 2007 the hardware platform was the Sony Aibo. Since 2008 the SPL hardware platform is the Aldebaran Nao humanoid (Fig.1). Its main sensors are two non-stereo cameras and with it the teams have been exposed to the complexity of biped movement.

Figure 1. Nao humanoid and Webots simulator

In SPL, the robot players must be completely autonomous. In order to build a soccer robot player many different habilities must be programmed, both perceptive and motion or control oriented. For instance the goto ball behavior, the follow-ball behavior, the ball detection, the kicking, self-localization, standing up in case of fall, etc.

This work is focused in the goal detection, based on the camera images of the Nao. The goal detection helps the robot to decide whether to kick the ball towards the opponent's goal or just turn to clear the ball out of its own goal. It can also provide good information to self-localization inside the game field.

The rest of this paper is organized as follows. Section II reviews the state of the art in artificial vision systems in the RoboCup. In section III several solutions to the same problem of goal detection in the images are proposed. Section IV proposes a technique for obtaining spatial information from the previously detected goals. Section V shows experiments with proposed techniques. Finally, conclusions and further improvements in given in section VI.

II. VISION BASED SYSTEMS IN THE ROBOCUP

Over the last years, considerable effort has been devoted to the development of Artificial Vision systems for the RoboCup

soccer leagues. In this way, the increasing competitiveness and evolution of the RoboCup leagues has conducted to vision systems with high performance, which are addressing a variety of typical problems [12], such as perception of natural landmarks without geometrical and color restrictions, obstacle avoidance, pose independent detection and recognition of teammates and opponents, among others.

Several constraints in the RoboCup domain make difficult the development of such vision systems. First, the robots always have limited processing power. For instance, in the Nao humanoid a single AMD Geode 500Mhz CPU performs all the onboard computations and the Naoqi middleware consumes most of that capacity. Second, the robot cameras use to have poor quality. In the Aibos the camera was of 416x320 pixels and the colors were not optimal. Third, the camera is constantly in motion, not stable in height as the robot moves through the field.

A. Background

A number of initiatives for developing vision systems conceived to give solutions to the aforementioned typical problems have been carried out in recent years. In this line, early work by Bandlow et al. [8] developed a fast and robust color image segmentation method yielding significant regions in the context of the RoboCup. The edges among adjacent regions are used to localize objects like the ball or other robots on the play field. Besides, Jamzad et al. [10] presented several novel initiatives on robot vision using the idea of searching on a few jump points in a perspective view of robot. Thus, they performed a fast method for reliable object shape estimation without the necessity of previously segmenting the images.

On the other hand, the work by Hoffmann et al. [11] introduced an obstacle avoidance system that is able to detect unknown obstacles and reliably avoid them while advancing toward a target on the play field of known color. A radial model is constructed from the detected obstacles giving the robot a representation of its surroundings that integrates both current and recent vision information.

Further vision systems include visual detection of robots. In this sense, Kaufmann et al. [13] proposed a methodology that consists of two steps: first, the detection of possible robot areas in an image is conducted and, then, a robot recognition task is performed with two combined multi-layer perceptrons. Moreover, an interesting method presented by Loncomilla and Ruiz-del-Solar in [12] describes an object recognition system applied to robot detection, based on the wide-baseline matching between a pattern image and a test image where the object is searched. The wide-baseline matching is implemented using local interest points and invariant descriptors.

Furthermore, recent work proposed by Volioti and Lagoudakis [14] presented a uniform approach for recognizing the key objects in the RoboCup This method proceeds by identifying large colored areas through a finite state machine, clustering of colored areas through histograms, formation of a bounding boxes indicating possible presence of objects, and customized filtering for removing unlikely classifications.

B. Related work on visual goal detection

In general, the aforementioned research has been oriented to solve visual tasks in the environment of the RoboCup. However, some of those works have specifically proposed solutions to the problem of goal detection.

In this regard, one of the earliest approaches was given by Cassinis and Rizzi [9] that performed a color segmentation method using a region-growing algorithm. The goal posts are then detected selecting the boundary pixels between the goal color and the white field walls. After that, image geometry is used to distinguish between the left and the right goal post.

The aforementioned work described in [8] has been also applied to the detection of the goals in the RoboCup leagues. In this way, they detect goals by the size of the regions obtained after applying the color based image segmentation mentioned above. Moreover, [14] aims at recognizing the vertical goal posts and the goal crossbar separately. Both horizontal and vertical goal indications and confidence levels are derived from the horizontal and vertical scanning of the images, according to the amount of lines detected. Afterwards, it is decided whether the previously obtained indications can be combined to offer a single goal indication and, finally, different filters are used to reject unlikely goal indications.

III. GOAL DETECTION IN 2D

Two different approaches oriented to the detection of the goals that appear in the 2D images are described in the next two subsections. The first one puts the emphasis on the geometric relations that must be found between the different parts that compose a goal, while the second is focused on edge detection strategies and specifically in the recognition of pixels belonging to the four vertices of a goal: Pix1, Pix2, Pix3 and Pix4 as shown in Fig.2.

A. Detection Based on Geometrical Relations

The first proposed method is intended to be robust and fast in order to overcome some of the usual drawbacks of the vision systems in the RoboCup, such as the excessive dependency of the illumination and the play field conditions, the difficulty in the detection of the goal posts depending on geometrical aspects (rotations, scale,...) of the images captured by the robots, or the excessive computational cost of robust solutions based on classical Artificial Vision techniques. The proposed approach can be decomposed into different stages that are described in the next subsections.

1) Color calibration: The first stage of the proposed method consists of a color calibration process. Thus, a set of YUV images acquired from the front camera of the Nao robot is segmented into regions representing one color class each.

Fig.2 shows an example image captured by the Nao robot containing a blue goal.

The segmentation process is performed by using a k-means clustering algorithm, but considering all the available centroids as initial seeds. Thus, in fact, seven centroids are utilized, corresponding to the colors of the ball (orange), goals (yellow and blue), field (green), robots (red and blue) and lines (white).

Figure 2. Example of original image from a RoboCup competition

Figure 3. Color segmentation

The range between the minimum and the maximum YUV values in the regions obtained after that clustering stage are considered as the actual prototype values that characterize each color class of interest. Fig.3 depicts the color image segmentation produced by applying the range of color values automatically obtained through the calibration to the example image in Fig.2. The good segmentation results in Fig.3 indicate that the prototype values for each color of interest have been correctly determined during the calibration process.

2) Geometral and Horizon Planes Detection: The next step consists of the estimation of the geometral and horizon planes according to the robot head position. In order to do this, firstly, the pitch and yaw angles that indicate the relative position of the robot head with respect to the play field are calculated. On the one hand, the geometral plane is defined as the horizontal projection plane where the observer is located. On the other hand, the horizon plane is parallel to the geometral plane and indicates the level above which there is no useful information.
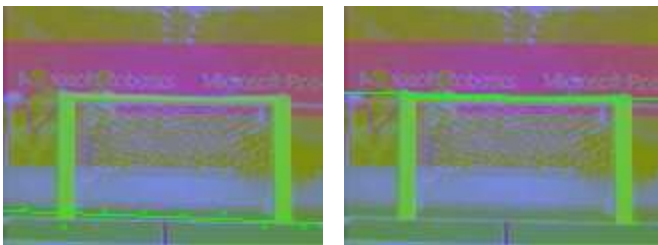

Figure 4. Intersection of geometral and horizon plane with image plane

Thus, the position matrix of the robot head is used for determining the horizontal inclination of the image with respect to the play field. Then, a grid composed of series of parallel vertical lines perpendicular to the horizontal inclination previously mentioned is calculated. The intersection between the grid and the green play field produces a set of points.

The line across these points is the intersection line between the geometral plane and the image plane. In fact, the goal posts will be searched above this line. Fig.4 (left) displays the intersection between the geometral plane and the image plane corresponding to the example image in Fig.2.

Furthermore, intersections among the grid and the top blue or yellow pixels in the image are detected (taking into account the inclination of the image). The line across those points constitutes the intersection among the horizon plane and the image plane. It is expected not to find useful information in the images above this line. Fig.4 (write) displays the intersection between the horizon plane and the image plane in the example image in Fig.2. Note that, by definition, the geometral and the horizon planes are parallel and delimit the region where the goals are expected to be found.

3) Goal Posts Detection: The overall aim of this process is to extract the goal posts and other interesting features that could reinforce the detection of goals in the play field.

First of all, the color prototypes obtained as explained in Section III.A1 are used to segment the blue and yellow goal posts and crossbars. In order to do this, not all the image pixels are analyzed, but a high resolution sampling grid is utilized in order to detect blue or yellow lines in the image. Fig.5 depicts the detected lines corresponding to a blue goal (long lines correspond to the posts and short blue lines to the crossbar) corresponding to the example image in Fig.2.
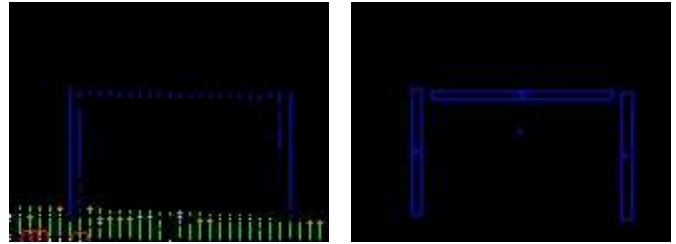

Figure 5. Interest points and goal blobs

In addition, a process to detect interest points is performed. The same grid mentioned before is utilized to detect crossings between blue or yellow lines (belonging to the goal posts) and white lines in the play field (goal lines). Also, crossings among green pixels (belonging to the play field) and white lines that delimit the play field are identified. If those interest points are detected close to the blue or yellow lines, previously sampled, they reinforce the belief that those lines belong to the goal posts. Red circles in Fig.5 (left) enclose interest points identified in the original image shown in Fig.2.

4) Goal Recognition: Once a set of pixels distributed into parallel lines corresponding to the goal posts and crossbar have been identified according to the procedure described in the previous section, the last step consists of a recognition process that finally locates the gravity center of the goal.

In order to perform such task, the aforementioned lines are grouped into blobs. A blob is composed of neighbor lines with similar aspect ratio (an example is shown in Fig.5 (right)). Finally, the blobs identified in this way are grouped into a perceptual unit that can be considered as a pre-attentive goal. Then, we apply an intelligent case reasoning strategy to bind

that unit into a coherent goal. Fig.5 (right) illustrates the blobs that configure the goal that appears in Fig.2 after it has been recognized by the proposed technique. The geometric center of the goal is also indicated according to the recognition method.

B. Detection based on color, edges and Hough transformation

We have also developed a second simple method to detect goals in 2D images. It follows four steps in pipeline. First, a color filter in HSV color space selects goal pixels and maybe some outliers. Second, an edge filter obtains the goal contour pixels. Third, a Hough transformation gets the goal segments. And fourth, some proximity conditions are checked on the vertices of such segments, finding the goal vertices Pix1, Pix2, Pix3 and Pix4. All the steps can be shown at Fig.6.
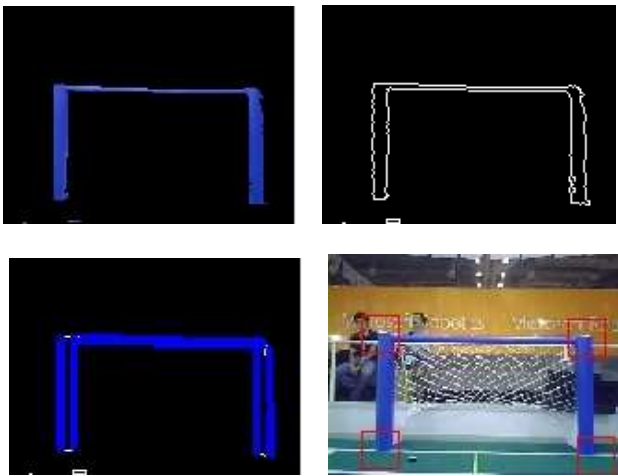

Figure 6. Goal detection based on color, edges and Hough transformation

IV. GOAL DETECTION IN 3D

Once the goal has been properly detected in the image, spatial information can be obtained from the that goal using geometric 3D computations. Let Pix1, Pix2, Pix3 and Pix4 be the pixels of the goal vertices in the image, which are calculated with the algorithms of section III. The position and orientation of the goal relative to the camera can be inferred, that is, the 3D points P1, P2, P3 and P4 corresponding to the goal vertices. Because the absolute positions of both goals are known (AP1,AP2,AP3,AP4) that information can be reversed to compute the camera position relative to the goal, and so, the absolute location of the camera (and the robot) in the field.

In order to perform such 3D geometric computation the robot camera must be calibrated. Its intrinsic parameters are required to deal with the projective transformation the camera does over objects in 3D world when it obtains the image. The pinhole camera model has been used, with the focal distance, optical center and skew as its main parameters. In addition, two different 3D coordinates are used: the absolute field based reference system and the system tied to the robot itself, to its camera.

We have developed two different algorithms to estimate the 3D location of the perceived goal in the image. They exploit different geometric properties and use different image primitives: line segments and points.

A. Line segments and thorus

Our first algorithm works with line segments. This algorithm works in the absolute reference system and finds the absolute camera position computing some restrictions coming from the pixels where the goal appears in the image.

There are three line segments in the goal detected in the image: two goalposts and the crossbar. Taking into consideration only one of the posts (for instance GP1 at Fig.2) the way in which it appears in the image imposes some restrictions to the camera location. As we will explain later, a 3D thorus contains all the camera locations from which that goalpost is seen with that length in pixels (Fig.8). It also includes the two corresponding goalpost vertices. A new 3D thorus is computed considering the second goalpost (for instance GP2 at Fig.2), and a third one considering the crossbar. The real camera location belongs to the three thorus, so it can be computed as the intersection of them.

Nevertheless the analytical solution to the intersection of three 3D thorus is not simple. A numerical algorithm could be used. Instead of that, we assume that the height of the camera above the floor is known. The thorus coming from the crossbar is not needed anymore and it is replaced by a horizontal plane, at h meters above the ground. Then, the intersection between three thorus becomes the intersection between two parallel thorus and a plane. The thorus coming from the left goalpost becomes a circle in that horizontal plane, centered at the goalpost intersection with the plane. The thorus coming from the right goalpost also becomes a circle. The intersection of both circles gives the camera location. Usually, due to simmetry, two different solutions are valid. Only the position inside the field is selected.

To compute the thorus coming from one post, we take its two vertices in the image. Using projective geometry and the intrisinc parameters of the camera, a 3D projection ray can be computed that traverses the focus of the camera and the top vertex pixel. The same can be computed for the bottom vertex. The angle α between these two rays in 3D is calculated using the dot product.
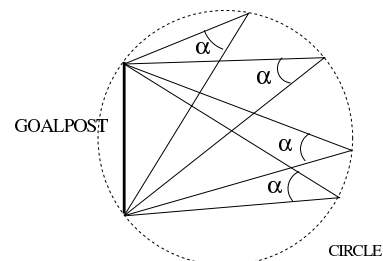

Figure 7. Circle containing plausible camera positions

Let's now consider one post at its absolute coordinates and a vertical plane that contains it. Inside that plane only the points

in a given circle see the post segment with an angle α. The thorus is generated rotating such circle around the axis of the goalpost. Such thorus contains all the camera 3D locations from which that post is seen with a angle α, regardless its orientation. In other words, all the camera positions from which that post is seen with such pixel length.
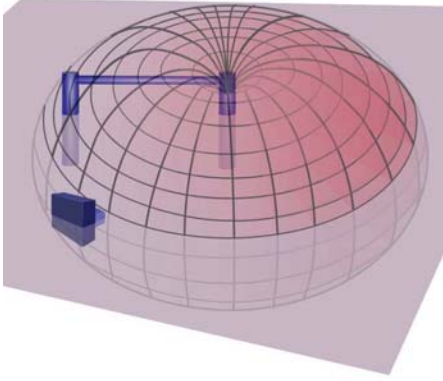

Figure 8.   Thorus containing all plausible camera positions

B. Points and projection rays

The second algorithm works in the reference system tied to the camera. It uses three goal vertex pixels Pix1, Pix2 and Pix3. For Pix1, using the pinhole camera model, a projection ray R1 can be drawn which traverses the camera focus and contains all the 3D points which project into such Pix1. R2 and R3 rays are computed in a similar way, as seen in Fig.9. The problem is to locate the P1, P2 and P3 points into their corresponding projection rays.

Assuming that we know the position of P1 in R1 then only a reduced set of points in R2 and R3 are compatible with the real goal size. Because the distance between P1 and P2 is known (D12), P2 must be in R2 and the sphere centered at P1 with D12 radius, named S2 (Fig.9). The general intersection between R2 and S2 yields two candidate points: P2' and P2" (there can also be no interesection at all or only one single point). Following the same development and the distance D13 between P1 and P3, two more candidate points are computed: P3' and P3".

Combining those points we have several candidate tuples $(P1, P2', P3')$, $(P1, P2'', P3')$, $(P1, P2', P3'')$ and $(P1, P2'', P3'')$ All of them contain points located at the projection rays and all of them hold the right distance between P1 and the rest of points, but the distance between P2 and P3 may not be correct. Only the real solution provides good distances between all of its points. A cost function can be associated to choose the best solution tuple. We used the error in distance between P2 and P3, compared to the good distance D23.

In fact, the P1 position in R1 is not known, so a search is performed for all the possible P1 values. The algorithm starts placing P1 at λ distance from the camera location. All the candidate solution tuples are calculated and their costs computed. For each λ the cost of its best tuple is stored. The
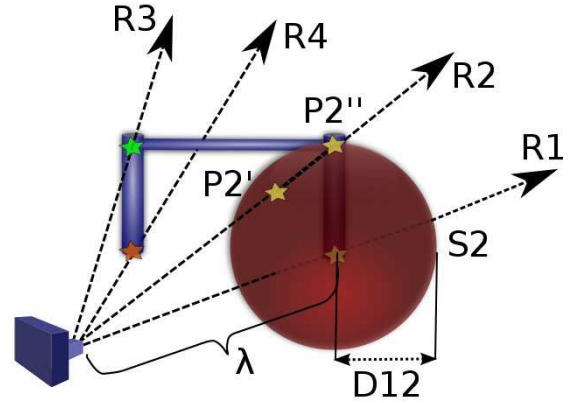

Figure 9.   Projection rays for the four goal corners

search algorithm explores R1 increasing λ at regular intervals of 10cm, starting close to the camera location and up to the field size, as can be seen at Fig.10. The tuple with the minimum cost is chosen as the right P1, P2 and P3 values. P4 is directly computed from them. They are the relative 3D position of the goal in the camera reference system.
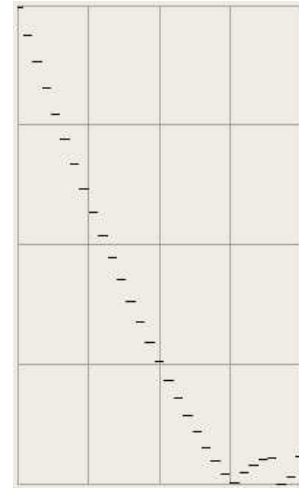

Figure 10.   Cost function for different λ values

Finally, the absolute 3D camera position can be computed from $(P1, P2, P3, P4)$. Because the absolute positions of the goal in the field reference system are known $(AP1, AP2, AP3, AP4)$, we can find a rotation and translation matrix RT that fits the transformation of P1 into AP1, P2 into AP2, etc. We have used the algorithm in [1] for that. The estimated translation represents the absolute position of the camera in the fied based reference system.

V. EXPERIMENTS

Several experiments have been carried out to validate our algorithms, both in simulation and with real images. For simulated images we have used Webots (Fig.1) and for real ones a benchmark of images collected from the Nao's camera at the RoboCup2008, placing the robot at different field locations.

The first set of results presented in this section correspond to the 2D goal detection strategy presented in Section III-A. In particular, Fig.11, Fig.12 and Fig.13 display three examples corresponding to real RoboCup images and the results produced by the different steps of the proposed method. Thus, right images in the first row of each figure show the intersection between the geometral plane and the image plane. The left image in the second row of each figure displays those pixels that are part of the goal according to the sampling grid utilized and, also, the interest points detected. Finally, right images depict the recognized goal and its gravity center for each example image. As it can be appreciated, the proposed strategy is able to recognize goals even in situations involving certain difficulties, such as when only a small part of a goal appears in the image, or if the play field is not visible in the image, or when the goal is seen from a side in the play field.
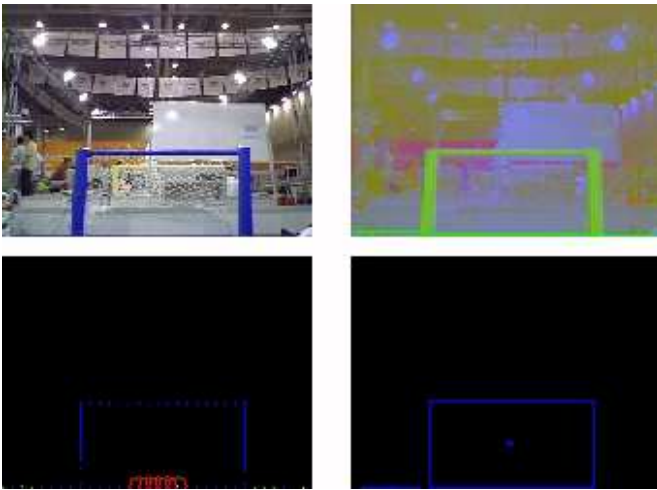

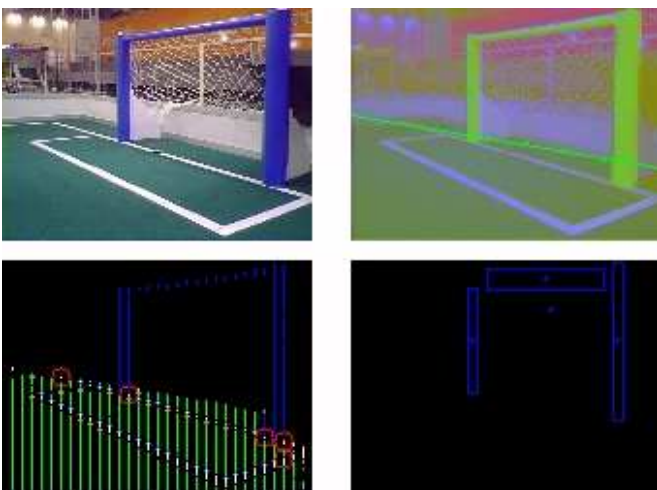Figure 11.   Goal detected with method described at Section III-A

Figure 12.   Goal detected with method described at Section III-A

A second set of results produced by the technique explained in section IV are shown in Fig.14. The input images are displayed at the right side. The output of the 2D
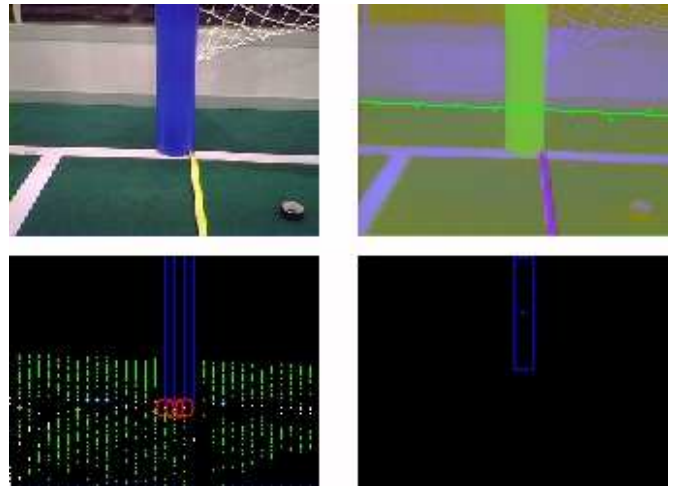

Figure 13.   Goal detected with method described at Section III-A

detection is overlapped as red squares at the goal vertices (Pix1,Pix2,Pix3,Pix4). In the left side of the figure the field lines and goals are drawn, and the estimated camera position is also displayed as a 3D arrow. No accuracy has been quantitatively measured yet, but the results seem promising and qualitatively correct.
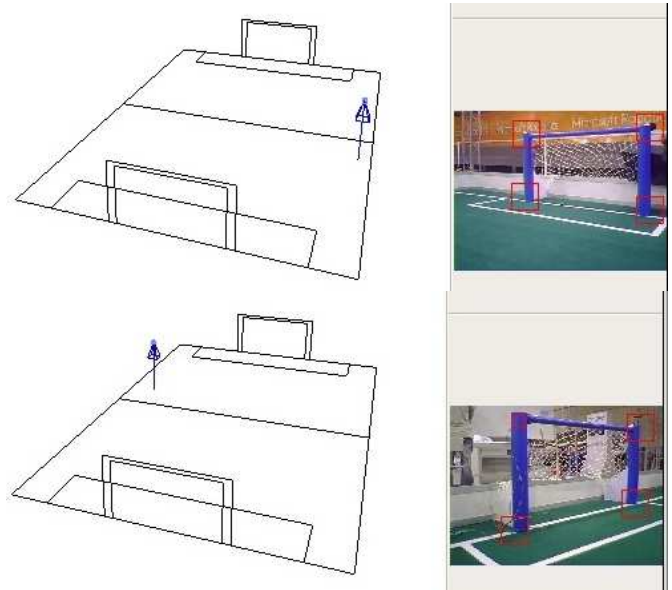

Figure 14.   3D position from the goal detected in the image

The Nao camera has been calibrated as an ideal pinhole camera, with its optical center at the middle of the image and its focal distance inferred from the hardware specifications. That's why no further accuracy test has been carried out yet.

All the results presented in this paper have been obtained on a 3 GHz Pentium-IV machine processing offline the real images. The time consumption corresponding to both 2D and 3D proposed techniques are shown in table I. In particular, times for each of the processing steps to detect the goal in 3D are shown. The 3D algorithm is fast. The projection rays

method is slower than the thorus method, maybe because it is a search algorithm. For edge filter and Hough transformation we have used OpenCV library.

2D detection based on Geometrical Relations	9ms.
2D detection based on color, edges and Hough transformation	13'2ms.
Thorus based 3D detection	1ms
Projective rays 3D detection	2ms.

Table I
TIME CONSUMPTION

The algorithm performs well both with the ideal images coming from the Webots simulator and the real images from the Nao at RoboCup2008. In the case of the 2D goal detection at III-B the color filter must be properly tuned for each scenario.

VI. CONCLUSION

Vision is the most important sensor of the autonomous robots competing at the RoboCup. Goal detection is one of the main perceptive habilities required for such autonomous soccer player. For instance, if the humanoid had the opponent's goal just ahead it should kick the ball towards it. If the goal in front of the robot was its own goal, then it should turn or clear the ball away. In this paper two different algorithms have been proposed to detect the goal in the images coming from the robot's camera. One of them based on geometral and horizon planes. The second one using a HSV color filter, an edge filter and Hough transformation to detect the post and crossbar lines.

In addition, two new methods have been described which estimate the 3D position of the camera and the goal from the goal perceived inside the image. The first one uses the line length of the posts and intersects two thorus to compute the absolute 3D camera position in the field. The second one uses the projection lines from the vertice pixels and searches in the space of possible 3D locations. This 3D perception is useful to the self-localization of the robot into the field.

All the algorithms have been implemented as a proof of concept. Preliminary experiments have been carried out that validate them and the results seem promising as shown in Section V.

We are working on performing more experiments onboard the robot, with its limited computer. We intend to optimize the implementation to reach real time performance.

The proposed 3D algorithms assume the complete goal appears in the image, but this is not the general case. The second future line is to expand the 3D geometry algebra to use the field lines and incompletely perceived goals as source of information. For instance the corners and field lines convey useful self-localization information too.

REFERENCES

[1] A. Lorusso, D.W.Eggert and R.B.Fisher, *A Comparison of Four Algorithms for Estimating 3-D Rigid Transformations*, Proceedings of the 1995 British conference on Machine vision, Vol. 1, pp 237-246, Itech Education and Publishing 1995.

[2] Thomas Rofer et al., *B-Human, Team Report and code release 2008*

[3] S. Lenser and M. Veloso, *Visual sonar: fast obstacle avoidance using monocular vision* in Proceedings of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, 2003. (IROS 2003), pp 886-891, 2003.

[4] M. Jüngel, *A Vision System for RoboCup*, diploma Thesis, Institut für Informatik, Humboldt-Universität zu Berlin, 2004.

[5] J.C. Zagal, J. Ruiz-del-Solar, P. Guerrero and R. Palma, *Evolving Visual Object Recognition for Legged Robots* in RoboCup 2003: Robot Soccer World Cup VII, LNCS 3020, pp 181-191, 2004.

[6] D. Herrero and H. Martínez, *Embedded Behavioral Control of Four-legged Robots* in Robotic Soccer, Pedro Lima editor, pp 203-227, 2007.

[7] RoboCup Standard Platform League (Nao) Rule Book.

[8] T. Bandlow et al., *Fast Image Segmentation, Object Recognition and Localization in a RoboCup Scenario* in M. Veloso, E. Pagello, and H. Kitano (Eds.): RoboCup-99, LNAI, vol. 1856, pp. 174-185, 2000.

[9] R. Cassinis and A. Rizzi, *Design Issues for a Robocup Goalkeeper*, in M. Veloso, E. Pagello, and H. Kitano (Eds.): RoboCup-99, LNAI, vol. 1856, pp. 254-262, 2000.

[10] M. Jamzad, E.C. Esfahani, S.B. Sadjad, *Object Detection in Changing Environment of Middle Size RoboCup and Some Applications*, Proc. IEEE Int. Symp. On Intelligent Control, Vancouver, Canada, pp. 807-810, 2002.

[11] J. Hoffmann, M. Jungel, and M. Lotzsch, *A Vision Based System for Goal-Directed Obstacle Avoidance used in the RC'03 Obstacle Avoidance Challenge*, 8th International Workshop on RoboCup 2004 (Robot World Cup Soccer Games and Conferences), LNAI, vol. 3276, pp. 418-425, 2005.

[12] P. Loncomilla and J. Ruiz-del-Solar, *Robust Object Recognition using Wide Baseline Matching for RoboCup Applications*, in RoboCup 2007: Robot Soccer World Cup XI, LNCS, vol. 5001, pp. 441-448, 2008.

[13] U. Kaufmann et al., *Visual Robot Detection in RoboCup Using Neural Networks*, in D. Nardi et al. (Eds.): RoboCup 2004, LNAI, vol. 3276, pp. 262-273, 2005.

[14] S. Volioti and M.G. Lagoudakis, *Histogram-Based Visual Object Recognition for the 2007 Four-Legged RoboCup League*, in Artificial Intelligence: Theories, Models and Applications, LNCS, vol. 5138, pp. 313-326, 2008.