An Efficient Need-Based Vision System in Variable Illumination Environment of Middle Size RoboCup

Mansour Jamzad and Abolfazal Keighobadi Lamjiri

Sharif University of Technology
Department of Computer Engineering
Azadi Ave. Tehran, Iran
jamzad@sharif.edu, keighoba@ce.sharif.edu

Abstract. One of the main challenges in RoboCup is to maintain a high level of speed and accuracy in decision making and performing actions by the robot players. Although we might be able to use complicated hardware and software on the robots to achieve the desired accuracy, but such systems might not be applicable in real-time RoboCup environment due to their high processing time. This is quite serious for the robots equipped with more than one vision systems.
To reduce the processing time we developed some basic ideas that are inspired by a number of features in the human vision system. These ideas included *efficient need-based vision*, that reduces the number of objects to be detected to a few objects of interest with the minimum needed accuracy, *introduction of static and dynamic regions of interest*, which proposes the most probable areas to search for an object of interest, *an experimentally reliable method for color segmentation in variable illumination situation*, and finally, the *usage of some domain specific knowledge* that is used in detecting and tracking a unique safe point on the ball.
We have implemented these methods on RoboCup environment and satisfactory results were obtained.

Keywords: RoboCup, Object Detection, Need-based Vision, Variable Illumination.

1 Introduction

Human vision system, including its ability to recognize objects is based on a combination of image processing, volitional interpretation of colors and shapes, according to a prior knowledge and beliefs. Some of the main capabilities of the human vision system can be listed as follows:

1. The attention of human vision can go towards a particular object or area in the scene, extracting detailed and precise information about it. For example, in real soccer, during the game, a player has *close-up looks* at the opponent goal when he is about to shoot, but in many other situations he has just rough estimates for objects' locations.

D. Polani et al. (Eds.): RoboCup 2003, LNAI 3020, pp. 654–661, 2004.

However, except for a few systems that consider the problem of visual attention, computer vision systems have to process the entire image (scene) to locate the objects of interest.
2. Usually, human vision is a *Need-based Vision* mechanism; i.e., in order to perform a certain action, we receive the necessary information from our environment.
3. Human vision system can perfectly adjust itself to correctly determining colors in different illumination levels. Designing a machine vision system with the same capability of humans is not trivial. Extensive research on this issue has been done, some of which are listed in the area of color constancy (e.g. [6, 7]).
   However, since in RoboCup 2003, the rules allow variable illumination, therefore, in this paper we describe a method for estimating field illumination and color segmentation accordingly.
4. Our vision system uses the information and natural/logical relation of events in the environment and continuously adds it to our knowledge. This process, plays a key role in our fast and efficient interpretation of the scene and understanding of the environment. If we could model and implement such a fast and extendable knowledge-base for the robot, then a robot can correctly recognize objects in similar situations. However, current knowledge-based robot vision systems are far beyond that of humans.

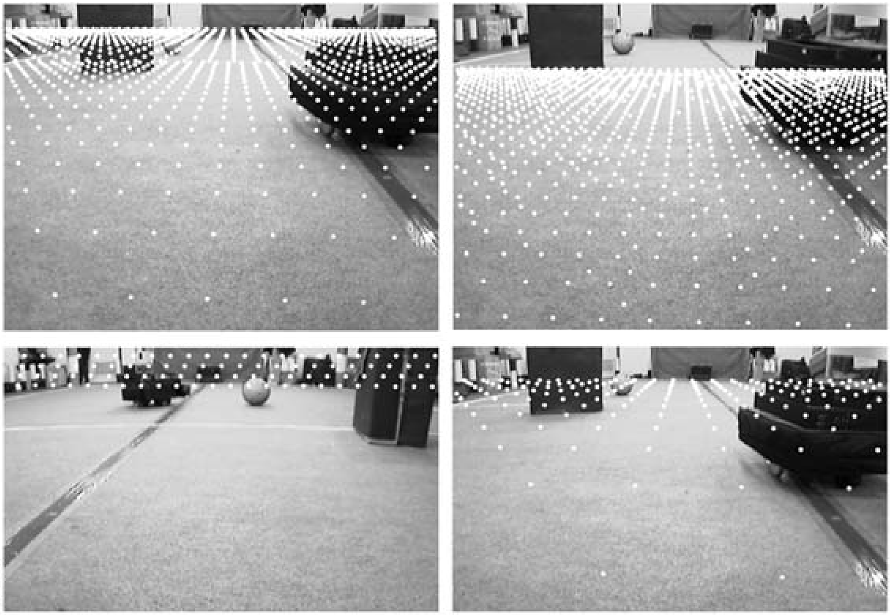
The mobile robots used in RoboCup usually have only one front view CCD camera that has a field of view of about 45 to 90 degrees, or an omni-directional viewing system that can view 360 degree around the robot. Our robots has both viewing systems ([1, 3]). Real-time processing of 30 frames per second is a challenging problem for the robots. In order to get a near real-time speed for the vision system, we introduce intelligent methods inspiring a number of the above mentioned features in human vision system. In our method a mobile robot can use its processing power efficiently by extracting only the required information with minimum needed accuracy in different situations.

2 How Can a Robot See Intelligently

In present stage of RoboCup research, a robot vision system may be considered intelligent if it can perform some degree of human intelligence used by a human player during a real soccer game. In the following sections we address a number of these features and propose methods towards exploiting them.

2.1 Static and Dynamic Regions of Interest

We have introduced a fast search mechanism based on an idea of *Perspective Jump Points* [1] for finding objects in the RoboCup field. In our previous work, we used only one pattern of jump points with a perspective distribution as shown in figure 1(a) for finding all objects in the field. These jump points are distributed in such a way that no matter where the ball (i.e. the smallest object) might be located in the field, at least 5 such jump points will be located on the ball. Although it is possible to find a minimum number of such jump points and their distribution, this optimal model would not be applicable because of existing shadows under, labels on, and brightness on top of the ball. So, we experimentally determined the distribution of all maps of jump points (figure 1), which are discussed later on.



a b
c d

Fig. 1. (a) Map of jump points for the front view CCD camera used for detecting the ball. (b) - (d) Map of jump points used for detecting goal objects, field lines and player robots, respectively.

Since in RoboCup each object has a unique color, by introducing the idea of jump points, we have reduced the problem of color segmentation of the whole image to the problem of examining the color of pixels at jump points followed by a novel idea of region growing [1].

However, in our way towards modeling the efficient need-based features of human vision for a robot, the robot vision system should be able to detect any object of interest in a least possible time. The areas on which we search to find a class of objects is called *Predefined* or *Static Regions of Interest (ROI).*

Table 1 and 2 show the frequency of time that the omni-vision camera and front view camera needed to use the static and dynamic ROI. For example, for searching the ball by omni-directional view camera, only in 40 % of times we need to use its static ROI that contained 3400 jump points (i.e. figure 1 (a)) but, in the rest (60 %) of time the Dynamic ROI was used to find the ball. Similar results are given for robots and also the field lines for front vision camera as well. The right most column of these tables show the processing time needed for object detection using jump points in static ROI.

We used a SONY VAIO laptop with Intel 800 MHZ CPU as the processor on each robot.

Table 1. Processing time needed for detecting objects using Static ROI in omni-directional viewing system.

Class of JPs	% of Usage	No of JPs	Detection Time(ms)
Ball	40%	3400	16
Robot	30%	500	5
Field Lines	Not Used	2600	11

Table 2. Processing time needed for detecting objects using Static ROI in front viewing system.

Class of JPs	% of Usage	No of JPs	Detection Time(ms)
Ball	20%	1100	160
Robot	10%	220	145
Field Lines	Not Used	1600	172

The idea of ROI, leads us to consider a separate map of jump points for each class of objects such as ball, goals, robots, field lines, etc. For example, if a goal is seen by a robot's front view CCD camera, it always appears at the topmost area of the image plane. Therefore in order to find the goal in this image we only need to check the color of jump points shown in figure 1 (c). Similarly the distribution of jump points for field lines and robots are shown in figure 1 (b) and (d), respectively.

However, we extended the idea of predefined search areas for objects (static ROI), and introduced *Dynamic ROI*. Suppose a robot is moving near the borders of the field or near the corner posts. In these situations, processing of a large sector of the omni-directional image that shows spectators and the background area would be a great waste of processing time. In these situations, Dynamic ROI is determined by using the *localization information* (that enables us to discard areas outside the field) or by *tracking of objects* in the field. As an example of the later case, we restricted search areas in the omni-directional image to just one half of the captured round image for the ball, when ball has been seen far in previous frames and it is most probable to appear in nearby locations. Tables 3 and 4 show a few Dynamic ROIs and the reduction percent in the number of search points obtained in omni-directional and front viewing systems, respectively.

2.2 Need-Based Vision

When we decide to do something and that job requires information from the environment, we start looking at specific objects to satisfy our needs. In addition, the information retrieved from the objects of interest may have low precision at first, and if higher resolution is required, we get that in close-up looks. In most current computer vision systems, usually there are a number of processing layers, each of which provides input data for another layer. The results are available when the topmost layer finishes processing its input data [4].

Table 3. Average decrease in number of search points in omni-directional vision system for Dynamic ROIs.

ROI Type	Usage	Decrease in No of JPs
Only Obstacles on the Moving Direction of Robot	60%	50%
Obstacle Tracking	60%	80%
Ball Within the Angle of its Previous Location	50%	70%
Ball Tracking	40%	95%

Table 4. Average decrease in number of search points in front vision system for Dynamic ROIs.

ROI Type	Usage	Decrease in No of JPs
Obstacle Tracking	50%	70%
Ball Tracking	50%	95%
Goal Tracking	80%	90%

However, inspiring need-based vision in human beings, we have developed a dynamically configurable vision system that is capable of providing just the needed information about the environment, i.e. detecting specific objects of interest, and specific precision of information. These requirements are provided by the playing algorithms of our robot. For example, while dribbling, the robot may need to get only the information about the nearest opponent player(s) in the direction of its movement with the highest possible accuracy.

Determining these sensing needs by playing algorithms of the robot can potentially be quite complex, as long as the robot can have a good degree of inter-agent communication and cooperation mechanisms. In this regard, suppose a teammate gets the possession of the ball; the sensory needs of other players will then change, that is, they do not need to look for the ball.

3 Using Domain Knowledge in Tracking the Ball

In RoboCup, the accuracy of the playing algorithms highly depend on the accuracy of ball detection, i.e. its distance and angle with respect to the robot. In this way, we need to know the (x, y) coordinate of ball's contact point with the field. This contact point is estimated as the middle point of the lower edge of a surrounding rectangle or bounding box around the ball [1].

3.1 Defining a Safe Border Point on the Ball

In practice, due to shadow under the ball and also the bright spots on top of the ball caused by projector illumination sources, in most cases, it is very difficult to find the bottom most and topmost points on the ball with relative accuracy. But in such situations, there is not much noise on the rightmost and leftmost parts of the ball. This means that these two parts can be detected more accurately.

Therefore, we introduced an algorithm for estimating the bounding box around the ball by detecting its leftmost and rightmost positions [2].

3.2 Tracking the Ball

Having determined a unique point on the border of the ball (i.e. the leftmost point on its horizontal diameter) we can compute the speed vector of the ball by detecting the same unique point in consecutive frames. Using this speed vector, we can track the ball by predicting the position of the bounding box around the ball in consecutive frames. As we discussed in section 2, this box is considered to be *dynamic ROI* for the ball. Therefore, to find the ball, we can first examine the area inside dynamic ROI and an area around it in the direction of speed vector. However, if the ball was not found, we can check the jump points of the static ROI of the ball (i.e. figure 1(a)).

4 Color Constancy in Variable Illumination

According to the rule changes of RoboCup 2003, variable illumination in a limited range is allowed[1]. In this situation, we have to consider the luminance of the field according which color segmentation shall be carried on.

4.1 Estimating the Field Illumination Level

In order to measure the field luminance, we mounted a piece of the same green carpet used for the field on top of robot body such that the omni-vision system of our robot can always see that. Figure 2(d) shows one of our robots with a green piece of carpet mounted on the right side of its body.

Since the green color of this piece of carpet will always be seen in fixed position in the image plane, we selected a small rectangle of size 10×30 pixels on the image plane corresponding to this piece of carpet. Field average illumination is estimated to be the average Y (i.e. Y component of YIQ color model [5]) value of all pixels inside the above mentioned rectangle.

However, the reason for using the green color as a reference for measuring the field illumination, is because the whole soccer field is covered with green carpet, and thus green is the dominating color in RoboCup environment.

4.2 Real Time Updating of Scene Illumination

In order to determine the scene illumination in real time, in each loop of our vision algorithm, we calculate the average intensity (i.e. $Y_{average}$) for a set of

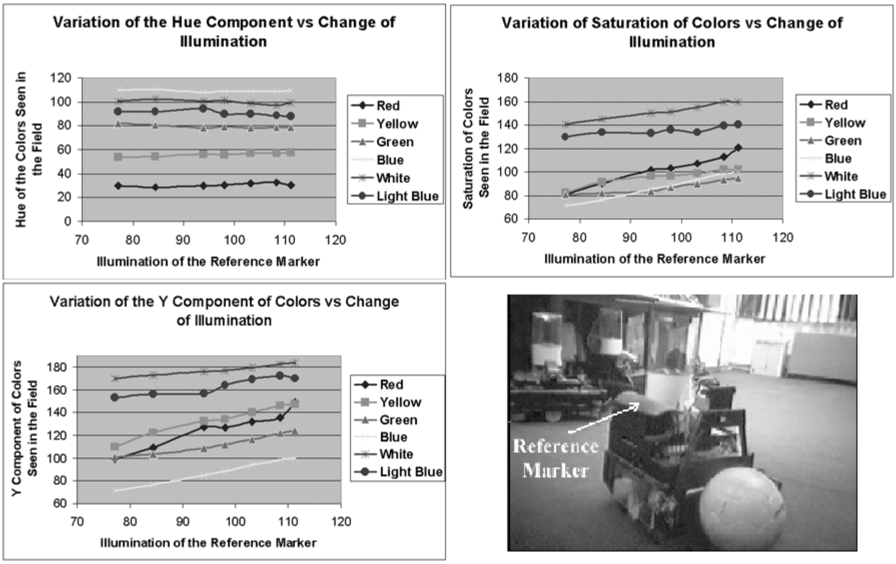
[1] From 800 to 1200 Lux.

pixels inside the green color marker. If this intensity is much different for its previously calculated value, a new average value is determined. In this way, the estimation for field illumination is updated upon changes in the environment illumination.

4.3 Using Average Illumination Estimate for Color Segmentation

For color segmentation we used the HSY color model (i.e. components $HandS$ are taken from HSI and Y is taken from YIQ color model [5].) Since the Y axis is perpendicular to HS planes, therefore, if the value of Y is known in an environment, then we can find the range for H, S and Y that best segment the standard colors in RoboCup. However, to give a solution for variable estimation situation, we used two set of lamps on the roof of our RoboCup field. One set of lights were always on, but we controlled the brightness of the other set in seven different levels. At each level, we manually selected a few areas on each standard color in RoboCup (i.e. red, green, yellow, etc) and determined the average values for H, S and Y for that colors. Figures 2 (a),(b) and (c) show these averages in each illumination level for H, S and Y.

Now, during the real games, since our vision system is dynamically updating the filed illumination (i.e. $Y_{average}$), by locating this $Y_{average}$ on the horizontal axis of figure 2, we can approximate the corresponding H, S and Y for each color and segment the image accordingly.



a b
c d

Fig. 2. (a)-(c) The average value of H, S and Y components in different levels of filed illumination for standard colors of RoboCup. (d) One of our robots having a piece of green carpet installed on it as reference marker.

5 Conclusion

Achieving the final goal of RoboCup that is "by year 2050 a team of robots will play against human world soccer champion" needs enormous amount of investment in related research fields, some of which can be named as mechanical design, accurate control, sensor fusion, multi agent cooperative behavior, etc., and especially a fast vision system.

In this paper, we introduced some basic concepts on how to develop an efficient vision system for robots in RoboCup environment. We introduced and implemented ideas of need-based vision, reduction of search areas in images acquired by robot's CCD cameras by means of static (predefined) and dynamic ROI, a color segmentation method in variable illumination. These methods were designed with the goal of reducing the processing time while maintaining a reliable level of accuracy.

Although our approach is in its early stages of making an intelligent robot vision system, but we believe that continuing research in this domain can lead to a level of intelligence that can be compared with that of human vision in a real soccer field.

Acknowledgments

We would like to thank a lot of the students in mechanics, hardware and software, who have worked hard in the past few years to construct the robots of Sharif CE team. In particular we would like to name Reza Zamninasab, Arash Farzan, Mansour Nurani, Mohamad Mahmoodi, Vahid Mirjalili, Reza Ghorbani and Moslem Kazemi. We are very grateful to all our sponsors and also we should give our sincere gratitude to all top ranking managers of Sharif University of Technology whose serious support made this work possible.

References

1. Jamzad M., et al., *A Fast Vision System for Middle Size RoboCup*, RoboCup-2001: Robot Soccer World Cup V, Peter Stone, et al (Eds), Springer, pp.76-85, 2001,
2. Jamzad M., Keighobadi Lamjiri A., *Towards an Intelligent Vision System for Mobile robots in RoboCup Environment*, To appear in Proc. of 2003 IEEE International symposium on Intelligent Control, Houston, TX, USA, Oct. 2003.
3. Jamzad M., et al., *An Omni-directional Vision System for Localization and Object Detection in Middle Size RoboCup*, Proc. of the IASTED International Conference, Applied Modeling and Simulation, Cambridge, MA, USA, pp.401-407, 2003.
4. Draper B., Hanson A., Riseman E., *Knowledge-Directed Vision: Control, Learning, and Integration*, Proc. IEEE Signals and Symbols, 84(11), pp.1625-1637, 1996.
5. Gonzalez R.C., and Woods R.E., *Digital Image Processing*, Addison-Wesley, 1993.
6. Finlayson G., Hordley S., Hubel P., *Color by Correlations: A Simple, Unifying Framework for Color Constancy*, IEEE Trans on Pattern Analysis and Machine Intelligence, Vol.23, pp. 1209-1211, 2001.
7. Funt B.V., Drew M.S., Ho J., *Color constancy from mutual reflection*, Int. Journal of Comp. Vision, Vol.6, pp.5-24, 1991.