SHAPE RECOGNITION AND ENHANCED CONTROL SYSTEMS FOR ROBOT SOCCER

Dr John Thornton[1], Mr. Joe Leonard[2], Mr. Richard Wisbey[3], Mr. Young Choon Lee[4] and Mr. Ken Spanks[5].

School of Information Technology, Griffith University Gold Coast, Parklands Dr, Southport, QLD 4215, Australia
Tel : 61-7-5552-8100; Fax : 61-7-5552-8066
E-mail: 1. j.thornton@gu.edu.au, 2. jleonard73@yahoo.com.au,
3. wiz2801@yahoo.com.au, 4. xtman1@hotmail.com, 5. forthoseofus@hotmail.com)

Abstract: *In this paper, we report on the current developments of our Mirosot Robot Soccer team, the "Robocoasters", in preparation for the FIRA Robot World Cup 2002 in Korea. We describe a number of unique enhancements to both the vision systems and the control system. Vision enhancements include a unique shape recognition system, expanded from recognition of three shapes to five shapes for the middle league competition. Control enhancements include improvements to our goalkeeper's behaviour, attacker and the introduction of role swapping between the robots. Finally, we outline our research underway and proposed future research for the team.*

Keywords: Shape Recognition, Vision, Control, Robot Soccer.

1. Introduction

This paper outlines the developments we have made to our Robot Soccer Team since the 2001 FIRA World Cup in China. Improvements are to both the vision system and the control system has been made. These improvements have been made in preparation for the 2002 FIRA World Cup in Korea.

The vision system involves a unique shape recognition system. This has recently been expanded from its original three shape recognition to identification of five shapes. The system provides faster and more reliable recognition of the robots during match play. An outline of the original system as well as a description of the extension made to the system is included in the paper.

The control system has improved in three main areas. Firstly, the attacking robot's (robot trying to score a goal) behaviour has been improved through implementation of a ball interpolation feature similar to that used by CMUnited in 1998 [1]. This feature was extended to include a varying velocity based on the robots current environment. This gave the robot speed when required to catch a target from a distance, but also maintains accuracy when having to turn acute angles.

The second improvement made to the control system has been improvements to the goalkeeper behaviour. The goalkeeper function is an extremely important role in the robot soccer domain as they provide the last defense against the opposition scoring a goal. An ineffective goal keeper may leave the team susceptible to both goals scored from the opposition as well as goals caused from the ball rebounding off objects or walls. Our goalkeeper's behaviours have been improved through the addition of new behaviours. This has been implemented in the form of a finite-state machine and has provided the team with an effective and reliable goalkeeper for match play.

The final improvement to the control system has been the introduction of a role-swapping behaviour. This was implemented in the system, firstly, to allow increase the probability of goal scoring and secondly to ensure the home goal is protected at all times. This is achieved through dynamically assigning the robots roles during play. An overview of the role-swapping behaviour is included in this paper as well as descriptions of factors that cause a swapping of roles

Finally we have included a brief conclusion, overview of both our present research and future proposals for research in this domain.

2. System Overview

The robots used by the Robot Soccer team are Yujin Robotics robots complying with the Mirosot rules. Until recently the "Robocoasters" have been a three robots side during match play, however, our team is moving from the Mirosot small league to the Mirosot middle league which provides for five robots per side during a match. It has been in preparation for this transition that much of the improvements to the system has been made.

The team uses a global vision system that features an overhead camera (Pulnix TMC-7DSP) that provides information to the vision system. The vision system creates a model of the environment, extrapolating information required by the control system. The information provided includes; coordinates of each object (home robots, opposition robots, ball etc), velocity of objects, and orientation of objects. The control system uses this information to determine the required velocity commands for the robot place the robot in the desired position. The frame grabber used (Matrox Meteor II) has a field acquisition rate of 60Hz (NTSC rate). This allows us to model the environment and send velocity commands 60 times per second.

The velocity commands are sent to the robot via an RF Module transmitter. This signal is received from a Radiometrix BiM Transceiver on the robot. The commands are translated into motor commands by Intel 80C296SA processors controlling each of the motors.. Both the vision and control systems are run on a work station with the following specification: Pentium III 933MHz processor, 512MB RAM, Windows 98. The development suite used was Visual Studio 6.0. Figure 1. provides an illustration of the entire robot soccer system. For further discussion of the rules of the Mirosot league refer to [2].

Figure 1. “Robocoasters” System

3 Vision System

The major attribute of our vision system that differentiates it from many other Robot Soccer teams is that colour and shape recognition is used. Alternatives to this strategy, which are often used, include colour recognition alone [3, 4, 5] and specialized hardware [6]. Our system is able to recognize home robots using only a single colour with each robot having that colour as a “cap” shaped as different geometrical shapes.

3.1. Prior Work

Our previous vision system [7] used three single coloured triangular shapes, one obtuse, one acute and one right-angled (See Figure 2). The algorithm used to identify these triangles, searches for the outlines of shapes consisting of our team colour (This colour is defined at initialization). We then construct all possible triangles that can be drawn around that outline using three straight lines, such that each line passes through at least two edge pixels without excluding a pixel from the shape.

After experimenting with various different algorithms that considered line length, angles and template matching we found the simplest most reliable heuristic was to accept the triangle with the smallest area. This accepted triangle is then matched to one of the actual triangles used on the robots. Once identified this triangle is then used to calculate the center point and orientation of each robot.

3.2. The New System

With the shift of our team from small league to middle league Mirosot we were required to develop a system that could recognize an additional 2 shapes. From a number of experiments in which recognition of additional triangles was attempted, we have found it is quite unreliable to get accurate vision information, identities and orientation using them. The reason for this is that the image data from the overhead CCD camera is somewhat susceptible to blurring and dulling of small shapes. Therefore the two additional shapes must differ greatly from the existing three. For this reason we have introduced two kite shapes. The shapes we have adopted along the original triangles are shown in Figure 2.

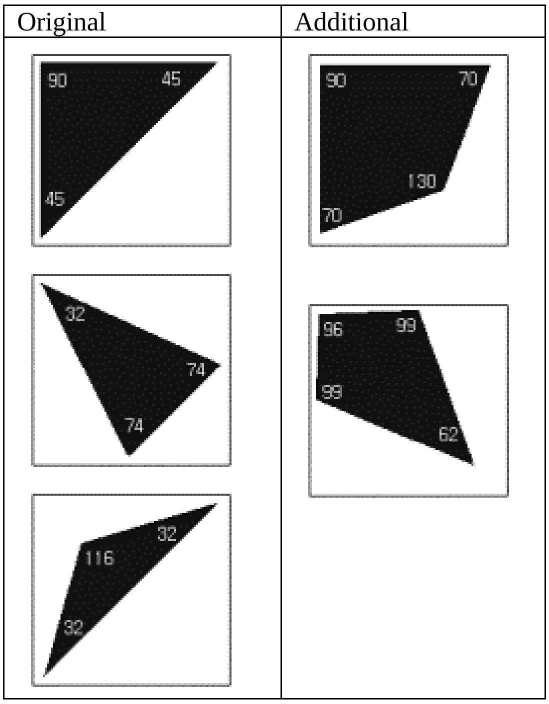

Figure 2. Shapes used by vision system

The algorithm for recognizing the kite shapes is, in fact, an extension and improvement of the previous triangle shape recognition algorithm. For all five robots we find the smallest triangles that will fit around the outlines (as in our previous system). We then separate the two biggest triangles among the five, to be processed as kite shapes. The same shape construction algorithm is used to the kite shapes with the only difference that four straight lines are drawn instead of three. Once the kite shapes are identified, adjustments are made so that only a portion of the kite shape is used (a triangle). By doing so, the existing orientation determination algorithm for triangles can be used without any changes.

In addition to the adoption of the kite shapes, an image enhancement algorithm has been implemented. This is the most recent enhancement to the vision system. It ensures the vision system is more effective, and can tolerate different lighting conditions, such as shadow, more reliably. With the image enhancement algorithm, we set two colour definitions of our team colour, a precise definition and an extended definition. The vision system, first, searches the pixels for the precise team colour. As soon as a certain number of pixels that suffice the criteria to determine it is one of the home robots, the extended definition of our team colour is applied to get a better image. The result has been a far more reliable vision system.

4. Goal Keeper

The goalkeeper's behaviour has been enhanced to provide for a more accurate and reliable player. This was achieved through the addition of a number of behaviours into the goalkeeper role. These behaviors are implemented in the form of a finite state machine. These behaviours give the goalkeeper sufficient behaviours to defend the goal in all conditions. The goalkeeper is the "Robocoasters" most reliable and effective player. Following is an overview of the Goalkeepers states and their entry and exit conditions. [7]

4.1. Align to Line

This state moves the Goalkeeper to a position on the goal line. The Goalkeeper enters this state when it is within a predefined threshold of the goal line and the projected path of the ball will not intercept the goal line. If the robots orientation is not inline with the goal line (excluding a small buffer), the state is re-entered and the robot re-aligns itself to the line. The state is only exited when the robot is inline with the goal line.

4.2. Patrol Line

The Patrol Line state is entered when the ball's position is in the home half of the pitch, the balls predicted intercept with the goal lone will intersect the goal line but not the goal mouth, and the goalkeeper is aligned to the goal line. Once entered the robot will patrol the goal line shadowing the lateral position of the ball. This state is exited when any of the entry conditions are broken.

4.3. Move to Position

The move to position state moves the robot into a defensive position on the goal line depending upon where the ball is. This state is entered on a number of different conditions but the reaction to each condition is different. The following is a list of each condition and action;

- *Condition*: Ball is in the opponents half:
  Action: Robot moves to the centre of the goal.
- *Condition*: Ball is in the quarter of the pitch nearest to the home goal, not in front of the home goal, and not predicted to go into the goal mouth
  Action: Robot moves to a point at the end of the goal mouth nearest to the ball.
- *Condition*: Ball is on either side of the second quarter of the pitch and not threatening the goal:
  Action: Robot moves to a point half way between the goal centre and the goal edge nearest the ball.

4.4. Intercept Ball

Intercept ball is a proactive defensive move, which is entered if the ball is on target for the goal and the ball is in the home half of the pitch. Once entered the robot will race as fast as possible to the predicted point on the goal line where the ball will intercept. This state is exited when the entry conditions are violated or changed.

4.5. Clear Ball

This state is entered when the ball is past the goal mouth and parallel with the side of the goal but not in the goal mouth (ie: to either side). Once entered the robot will attempt to 'kick' the ball away from the side of the goal by accelerating at maximum speed until the ball is struck or conditions change.

4.6. Spin Kick

The Spin Kick state is entered when the robot is stationary and about to intercept the ball. Once this state is entered the robot attempts to clear the ball by spinning on its axis and striking the ball to the side of the goal. The state exits when the ball is no longer going to be intercepted by the robot.

5. Attacker

5.1. Ball Interpolation and Velocity Variance

The attacking robot (the robot assigned to attempt a kick at goal) uses a ball interpolation behaviour using a varying velocity. The original ball interpolation behaviour was an implementation of the CMUnited 1998 Robocup teams attack algorithm as described in [1]. This includes an algorithm starting with a base velocity command for both wheels and adjusts the differential between these commands based upon the robots orientation, angle to the ball and the balls angle to the goal. This differential velocity is calculated every field steers and drives the robot to a target position. By using a changing target position [1] the robot can be steered behind the ball and then aimed at the goal to score a goal.

This above behaviour was implemented and the robot performed well but some problems were observed. The domain requires a controller to drive the robot swiftly while maintaining accuracy in order to score goals. When the behaviour was implemented initially, a high base velocity was chosen. This gave the robot the swiftness required to get to the ball, but accuracy, especially on tight angles, was sacrificed. When the controller was slowed the robot was too slow to reach the ball. A varying base velocity was introduced to allow the robot to attain the ball quickly if it were far away, but also maintain accuracy when required to turn tight angles.

The implementation of this variable velocity is as follows. An adjustment is now made to the base velocity based prior to being processed by the adjusting algorithm. Adjustments are made based on the robots distance to its target and the angle required to turn to the target. Examples of situations can be seen in Figure 3. Large distances to the target provide a faster base velocity. (see Figure 3b) This adjustment was made to ensure that in cases where the attacking robot is far from the ball, the player must get close to the ball as quickly as possible. The velocity is then further adjusted for the size of the angle difference between the robots orientation and the robots angle to the target. Large angles are given a slower base velocity (see Figure 3a), and small angles are given a faster base velocity. Sharp turns are quite often made when the robot is turning behind the ball to face the goal, by adjusting the velocity to slow when turning sharp angles, the robot maintains accuracy when attaining the ball. Should the robot get into a position that it is facing the goal and has attained the ball, velocity is maximized (see Figure 3c). This was introduced to ensure that the robot is given every opportunity to score a goal, and minimize the opportunity for opposition to intercept the ball.

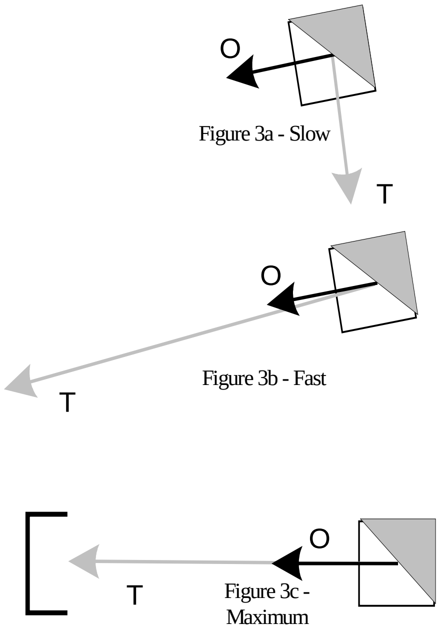

Figure 3a - Slow

Figure 3b - Fast

Figure 3c - Maximum

Figure 3. Situations for variations in velocity
O – Orientation of Robot
T – Target

6. Role Swapping

Robots are assigned one of three roles at the start of a match; goalkeeper, defender or attacker. During a match it may become more efficient for a player to take on a different role than that assigned at the start of play, that is, it is more efficient and effective for the robot to swap its playing role with another robot. Examples of instances when this may occur include; loss of recognition of a player (e.g. the goalkeeper is no longer recognized due to being knocked over during a match) and another robot being in a better position to score a goal than the current attacking robot. To overcome the problems associated with these situations a role swapping behaviour has been implemented in the team.

The role swapping behaviour has been implemented as a finite state machine and uses each robots current position, orientation, role and potential for play and uses these factors to assign each robot a role. The roles for each robot are assigned every time the environment (i.e. robot positions, orientations etc) is updated (60Hz). The roles are allocated in the following order; goalkeeper, attacker, defender.

6.1. Goal Keeper

The robot taking on the role of goalkeeper behaves as described previously in this paper. It is the first role assigned to a robot as this role was considered by the team to be imperative. If the goalkeeper has been lost by the vision system, such as if the robot is flipped over, the defender takes over the role of the goalkeeper. The defender was chosen to be the first to take over the role, as they are the closest to the goal line. Should the defender be lost by the vision system and be required to take over the goalkeepers role, the attacker retreats to its home goal line and takes over the role of goalkeeper.

6.2. Attacker

Once the system has chosen a robot to take on the role of goalkeeper, an attacker is selected. The attacking role is selected second as this will enable the team to both defend their goal and score goals with only two robots in play. The attacker will perform the behaviours described previously in the ball interpolation section of the paper. The attacker role will be assigned to either the remaining robot, if any robots have been lost by the vision system, or the robot able to reach and control the ball quickest. This is determined by not only the robots distance to the ball but also the balls travel direction and velocity and the robots current orientation.

6.1. Defender

The defender is the last role assigned by the system. The defender behaviour will only be assigned upon the other two roles being assigned. The defender performs the same role as the goal keeper except that its patrol line is midway between the home goal line and the half way line.

7.Conclusion and Future Research

The future of the Robocoasters control and vision systems will lie in the area of Artificial Intelligence proper. Research is currently being conducted into the use of a combination of reinforcement learning and fuzzy logic controllers for the control system. The research on the vision system is currently in the area of an ecological approach to perception, following the work of Gibson [3].

8. References

[1] Manuela Veloso, et al, "RoboCup-98 Small-Robot World Championship Team" 2000, Computer Science Department, Carnegie Mellon University, Pittsburg, PA, USA, July 1999.

[2] Mirosot rules, http//fira.cqu.edu.au/rwc2000/mrules.htm, 2001

[3] S. Subramanyam, P. Vadakkepat, K.C. Kim, J.H. Kim, “Multi-Agent Centralized Control in Soccer Robots”, *Proceedings of the Micro-Robot World Cup Soccer Tournament*, J.H. Kim and J.J. Lee (eds), KAIST Taejon, Korea, June 1997.

[4] R. Xiong and J. Chu, “A Global Vision System: Using The Hue Thresholds to Exact Feature and Recognize”, *Journal of Harbin Institute of Technology (New Series)*, Vol. 8, No. 3, 2001.

[5] R. Garcia, J. de la Rosa, A. Oller, A. Figueras and J.A. Ramon, “A Fast Vision System for Tracking in Robotic Soccer”, *FIRA World Cup France ’98 Procedings*, R. Stonier and J.H. Kim (eds), Paris, France, June 1998.

[6] M.F.M. Campos, M.M. Anício, M.A.S. Carvalho, R.M. Dias, A. Hartman, D.A.P. Nagem, V.H.A. Oliveira, E.S. Oliveira, G.A.S. Pereira, A.F. Ribeiro, F.C. Sanches and M.A. Silveira, “MIneiROSOT – The Development of a Centralised Control Set of Soccer-Playing Micro-Robots”, *FIRA World Cup France ’98 Procedings*, R. Stonier and J.H. Kim (eds), Paris, France, June 1998.

[7]Dr John Thornton and Valnir Ferreria Junior, "Towards a Behavior-Based Robotic Soccer Architecture" 2001, School of Information Technology, Griffith University, Unpublished.

[8] J.J. Gibson “The ecological approaoach to visual perception”. Boston: Houghton Miflfin, 1979.