FIRA Robot World Congress 2003
October 1 – 3, 2003, Vienna, Austria

CIIPS GLORY: A VISUAL SERVOING APPROACH WITHIN A BEHAVIOUR BASED FRAMEWORK FOR SOCCER ROBOTS

Jarrod Bassan, Joshua Petitt, Thomas Bräunl

Mobile Robot Lab
Centre for Intelligent Information Processing
School of Electrical, Electronic and Computer Engineering
The University of Western Australia
http://robotics.ee.uwa.edu.au
bassan@iinet.com.au, petitj01@mech.uwa.edu.au, braunl@ee.uwa.edu.au

Abstract

This paper describes the implementation of visual servoing techniques within a behaviour based system on a mobile robot. The visual servoing technique implemented on the robots is fast, reactive and robust, and appropriate for a highly dynamic environment, such as robot soccer. A simple method of calibrating the vision system, based on experimental data, was developed. A closed loop system was implemented to control the robot. The goal was to make the control system *model independent* and *predictive*, which involved two variations to the original design. Several other behaviours, essential to the task of playing soccer, have also been implemented on the robots. The implementation of behaviour based techniques has been successful and is demonstrated to be a viable method of controlling robots in dynamic environments.

Keywords: Robosot; Visual Servoing; Behaviour Based; Image Processing; Vision Calibration

1. Introduction

The CIPPS Glory 2003 team aims to demonstrate the viability of behaviour based techniques for controlling robots in highly dynamic environments. The context of the environment is a soccer games between robots where a robot must perform the task of moving the ball into a goal situated at one end of the field (ie; score a goal). The robot's success at its task will be largely governed by the speed at which it can act and react within its environment. To this end, the team has concentrated on implementing simple but robust behaviours which do not require complex processing.

The CIIPS Glory team uses a simple two-wheeled mobile robot platform, controlled by an EyeBot Mk IV controller [2] [4]. Each robot has several onboard sensors and actuators, including a digital compass, a colour camera, three proximity sensors and two driving motors for differential steering. The EyeBot controller is based on a Motorola MC68332 microcontroller, and is used both for image processing and for controlling the actuators [3] [5]. Sophisticated image processing is not possible (the EyeBot runs at 33 MHz clock speed), unless the frame rate or image resolution is reduced. Reducing the frame rate is not desirable because fast moving objects, such as the ball, cannot be tracked. The image resolution used is 80 by 60 pixels, and reducing image resolution below this level is not practical because the target will not be visible at long range. Therefore a simple, robust method based on colour information was developed, this is discussed in depth in section 3.1.

Figure 1 CIIPS Glory robot soccer team

2. Control Approaches

2.1. Behaviour Based Approach

The project aims to implement, on an autonomous robot, a set of behaviours that give the robot the abilities required to play soccer. The combination of behaviours will give the robot an 'instinct' to chase a ball, score goals, and avoid obstacles [18]. The combination of these behaviours will appear to an outside observer as if the robot is intelligently playing soccer [1].

To achieve the aim of scoring a goal, the robot must, as a minimum, exhibit these behaviours: search for the ball; chase the ball; control (dribble) the ball; turn in the direction of the goal; push or kick ball toward the goal; avoid obstacles en-route to goal. Each behaviour poses a separate problem, of which the ball-chasing behaviour is the most interesting, and has been implemented using visual servoing techniques, which is discussed at length in Sections 4.

2.2. Mind Framework

The Mind [16] framework was inspired by Sigmund Freud's model of the human mind [7]. Freud conceptualised a psychological model for the human mind composed of three components, the id, ego and super-ego. The id comprises one's basic instincts and resides completely in the subconscious. The ego performs conflict resolution between our basic instincts and our beliefs, moral or logical. The ego lies in both the conscious and subconscious regions of our mind. The super-ego, or the ego-ideal, is the mental picture one has of one's self and the "ideal" self. This is what Freud attributes to our concept of right and wrong.

Freud's model has been adapted to the field of computer science and forms the basis for a software framework, written in C++, which is capable of producing intelligent behaviour. The Mind architecture is composed of three 'layers', the Id, Ego and Superego, which perform functions analogous to their psychological counterparts. The Id is responsible for the processing and execution of Behaviours. A Behaviour is very similar to the classic perception neuron model, however the execution of behaviours is time-dependant, whereas the execution of a neuron model can be thought of as near instantaneous. This time dependence arises because all I/O operations are handled as behaviours; this has the advantage of abstracting any specific hardware access functions or pre-processing steps.

The implementation allows behaviours to be 'linked' together in a network, and signals are passed between Behaviours. Behaviour specific data is also passed via the links, however in general a Behaviour only accepts one type of data link for the input and another type for the output. Thus, the signal and data processing performed by individual behaviours are often very similar to classical feedback control methods.

2.3. World Model Approach

The traditional or classical approach to implementing a mobile robot is to construct a robot with several sensors, including a vision system. The robot uses all available sensor inputs and analyses them, combining all available data to produce a 'model' of the real world in 3D co-ordinates, and locates itself within this world. The robot then analyses its model, and makes decisions about its actions based on its model [8] [12].

There are several drawbacks to this approach. First, a large processing load is introduced because the robot must re-construct a 2D visual image into a 3D model. Second, the robot must maintain a 3D model of the world and know its location. Thirdly, the system is not inherently robust because errors can easily accumulate. For example, an incorrect identification of a landmark results in an incorrect 3D model, which leads to incorrect decisions [9] [10].

2.4. Visual Servoing Techniques

Robot soccer requires a robust target-tracking scheme that will control the robot such that it follows the ball (target) in a highly dynamic environment [6] [13]. Therefore it is desirable that the visual servoing technique be *model independent* and *predictive*. Model independent visual servoing control implies control without precise kinematic or camera (optical) models [16]. Predictive visual servoing means the control scheme has some ability to predict the future position of the target, based upon recent history.

3. Image Processing

3.1. Ball Identification

The first problem was identification of the ball within a single image frame, which is an image-processing task. It is known *a priori* that the target (ball) is a distinct yellow-green colour (tennis ball). Therefore an algorithm using colour identification was implemented

Pixel colour is determined using a process similar that proposed by Maxwell et al [14] for facial recognition on mobile robots. Maxwell's uses Red, Blue, Green (RGB) pixel values and 'fuzzy histograms' to identify skin colour in the scene, without converting to HSI (Hue Saturation Intensity). The 'fuzzy histogram' for red, green and blue pixels (r, g and b) is given by:

$$r = \mathrm{R} / (\mathrm{R} + \mathrm{G} + \mathrm{B})$$
$$g = \mathrm{G} / (\mathrm{R} + \mathrm{G} + \mathrm{B})$$
$$b = \mathrm{B} / (\mathrm{R} + \mathrm{G} + \mathrm{B})$$

The use of the 'fuzzy histogram' overcomes problems associated with lighting and shadow effects, and also avoids the need to convert the entire image into HSI (Hue Saturation Intensity) values.

The algorithm can be easily modified to scan every second row and/or column, or to only scan a portion of the image, thus reducing the processing load per frame and effectively increasing the maximum frame rate. Currently, a sustained frame rate of 4 frames per second has been achieved, therefore the robots can theoretically see an object moving at 1.4 m/s at a range of 0.5m.

The accuracy of the algorithm was further enhanced by also using pixel *intensity* (intensity = R + G + B) as a further cue for identifying the ball-pixels, as proposed by Spengler and Schiele [17].

3.2. Camera Calibration

The visual servoing control scheme uses a local polar coordinate frame of reference, and the ball-identification algorithm returns an (*x*,*y*) co-ordinate in the image frame. A mapping function is therefore required to map image Cartesian co-ordinates to polar co-ordinates relative to the robot. This was developed experimentally and negates the need to understand the optical properties of the camera. The method yields results that are sufficiently accurate for the robust visual servoing algorithm used to track the target. Importantly, the mapping function is fast and does not increase the processing load on the controller. The downside of this approach is that it assumes the shape and size of the target is uniform, and the robot is operating on a flat planar surface, and therefore the method is not appropriate for more general applications.

3.3. Experimental Results

The vision calibration experiment was conducted on a rectangular grid. The robot was located at the origin of the grid, such that the centre of the kicker mechanism was located at coordinates (0, 0). The target ball was placed at each grid intersection and the raw-image coordinates (as measured the aforementioned detection scheme) were recorded. The results were analysed to determine if a strong relationship between raw-image co-ordinates (x, y) and robot-centric polar co-ordinates (r, θ) could be found. The graphs in Figure 2 and Figure 3 clearly shows the correlations between x and θ, and y and r.

After analysis of the results, the following equations were chosen to map the image Cartesian coordinates (*x*,*y*) into robot polar coordinates (*r*,*q*)

$$r = \begin{cases} \dfrac{538}{y} & \text{for } y > 25 \\ 164\,y^{-0.06} & \text{for } y \le 25 \end{cases}$$

$$\theta = 0.6\,x - 25.5$$

The equation for *r* involves a piecewise function, which is an approximation to the actual function determined from the experiment; the exponential equation was implemented using a lookup table to increase speed.

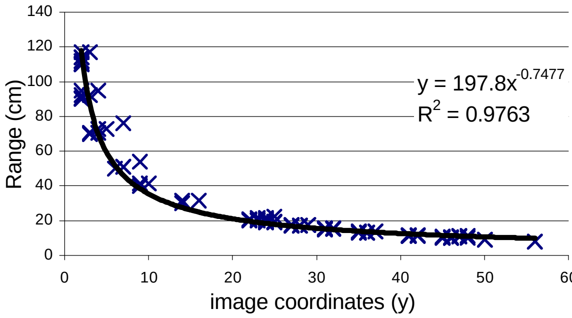

Figure 2 Correlation of image coordinates to object distance

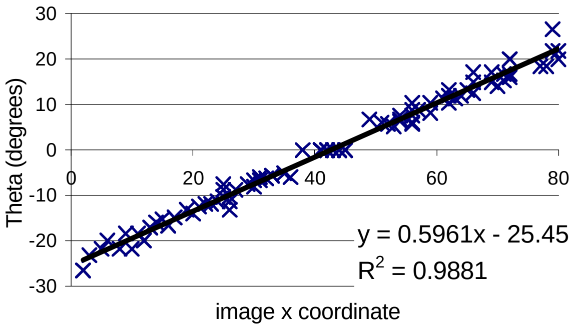

Figure 3 Correlation of image and object coordinates

4. Visual Servoing

The robot is controlled by two driving wheels, each connected to a direct current (DC) motor. A PI control algorithm, implemented as a Behaviour, is used to control the velocity of each wheel. The two wheel controllers receive signals from another Behaviour that controls the velocity of the vehicle as a whole.

4.1. Closed Loop Controller

A PID controller was implemented in software to control the robot rotation, such that the robots orientation was facing the ball.
Figure 4 shows a representation of the control algorithm, where C(s) is a PID controller implemented on the EyeBot controller, G(s) represents a model of the drive motors, and H(s) represents the target identification algorithm, which was calibrated such that H(s) = 1.

After adjusting the gains, the robot was able to satisfactorily track the target. The robot was made to 'chase' the target by summing a positive voltage with the output of the PID controller (C(s)), thus superimposing a forward motion with the rotation.

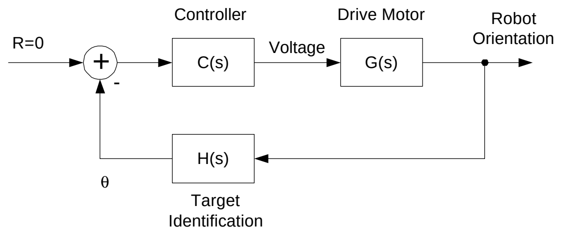

Figure 4 PID controller

The system was limited because the control-loop updated the motor voltages at an interval of every 40 ms, but the ball-position was only updated every image frame (approximately every 250ms). In terms of Figure 4, C(s) and G(s) were calculated at an interval of 40ms, but the output of H(s) was updated every 250ms (due to CPU limitations for image processing). As a result, the robot experienced significant over-shoot, even after adjusting the gains [11]. The problem was overcome by adding a 'compensator', which estimated that the ball angle had diminished (by some constant factor) for each iteration of the loop. The premise behind the compensator was a reasonable assumption, because the robot is expected to be turning towards the ball (thus the angle between the ball and robot axis diminishes). The addition of the compensator solved the overshoot problem, but the robot now responded poorly to moving targets.

Figure 5 is not strictly accurate in that sense of control theory, but the diagram is sufficient to convey the design of the system, including the compensator, represented as the function G'(s).

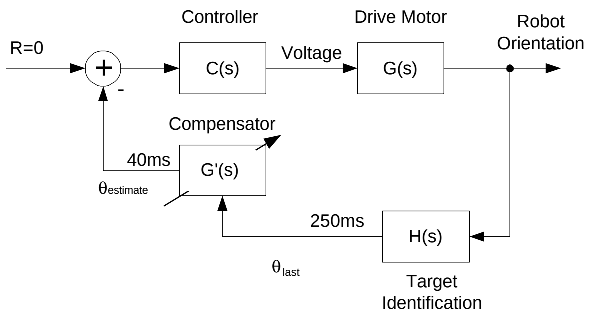

Figure 5 Controller with estimator

4.2. Predictive Approach

To improve the robots ability to track moving targets, a predictive approach was used. The control loop now includes a second controller (the 'predictor') that attempts to predict the position of the target (ball) one frame in the future. The compensator function was changed to interpolate between the last measured target position and the estimated position of the target in the next frame. In Figure 6 the predictor is represented as C'(s).

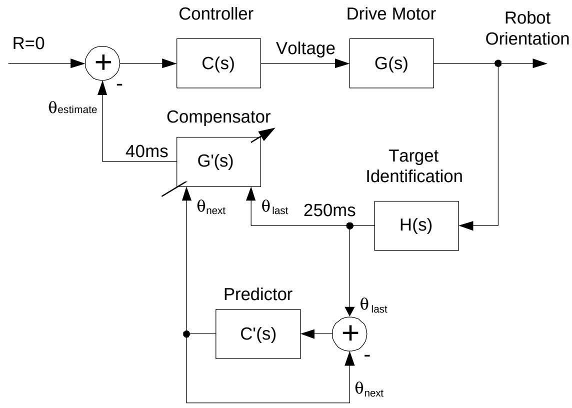

Figure 6 Predictive controller

4.3. Experimental Assessment

An experiment was devised to provide a qualitative assessment of different target chasing behaviours. The experiment was conducted by placing the robot in starting position. A ball was rolled in front of the robot, at a distance of 30cm. The velocity of the ball was 0.3m/s, and its path was perpendicular to the orientation of the robot. When the robot identified the ball, the visual servoing algorithm was activated and caused the robot to chase the target. The robot was configured to record the number of image frames in which the robot could identify the ball. The longer the robot was chasing the target, the more image frames the target would be identified in, thus the frame-count is a suitable indicator for how well the robot is able to chase the target.

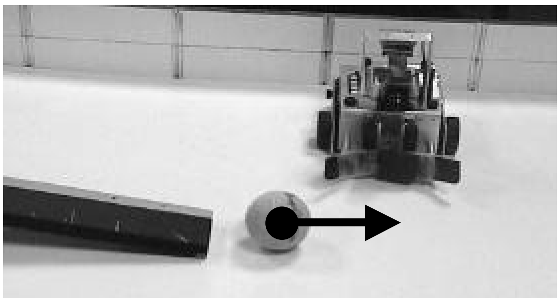

Figure 7 – Experiment to assess target tracking behaviour

The result presented in Figure 8 and Figure 9 compare the PID controller and simple estimator method against the predictor / estimator method, as discussed in sections 4.1 and 4.2. The graphs show the number of test-shots for which the robot was able to chase the target for a certain number of frames. The results demonstrate that the predictor / estimator method is able to track the target for longer periods of time, and is able to generate more successful target 'locks' – where a target lock is defined as tracking the target for more then 3 frames. The result supports the hypothesis that the predictor / estimator method is a superior method for tracking a moving target.

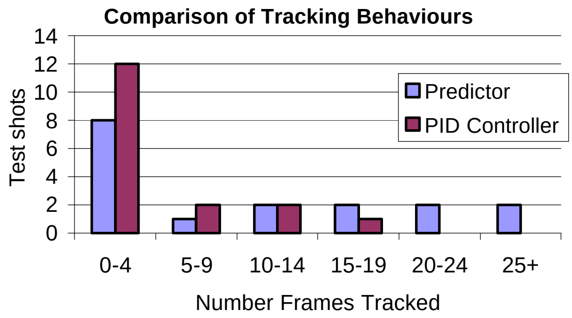

Figure 8 Target tracking comparison

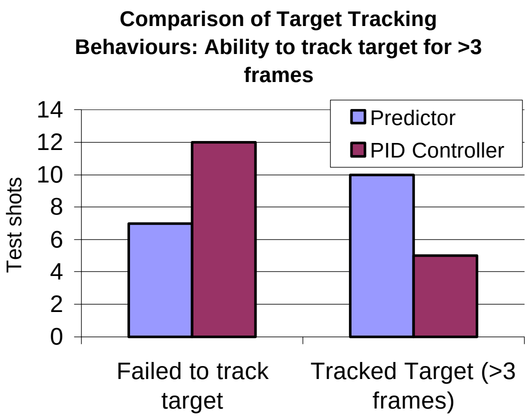

Figure 9 Number of continuous frames in target tracking

4.4. Goal Location

Rather then utilising the vision system to identify the goal, the robots use on board digital compasses to identify the goal-end of field. When the robot has control of the ball, it uses a PID controller to orient itself toward the goal-end and moves the ball in this direction. Since the goal-mouth is 33% of the width of the field, our method is expected to be on target for 33% of goal strikes.

5. Conclusion

The project has demonstrated the viability of all the behaviours required to give the robot the appearance of playing soccer intelligently. Some of these behaviours require further development and refinement. The main successes have been the development of a suitable vision calibration method and the development of a target-chasing controller that uses visual servoing techniques with a predictive approach, allowing the robot to chase a moving target.

The limited processing power of the controller resulted in some challenges to the implementation of the closed-loop controller, but the team has achieved its goals of building a system that is fast and reactive. Even with the limited image processing capabilities of the controller used, the behaviour based and visual servoing techniques have been successful. Future implementations on more powerful hardware are expected to yield even better results.

6. References

[1] R. Arkin, *Behavior-Based Robotics*, MIT-Press, Cambridge MA, 1998.

[2] T. Bräunl, *Embedded Robotics*, Springer-Verlag, Heidelberg, 2003.

[3] T. Bräunl, B. Graf, "Small Robot Agents with On-Board Vision and Local Intelligence", *Advanced Robotics*, vol. 14, no. 1, 2000, pp. 51-64.

[4] T. Bräunl, P. Reinholdtsen, S. Humble, "CIIPS Glory Small Soccer Robots with Local Image Processing", P. Stone, T. Balch, G. Kraetzshmar (Eds.), *Robocup 2000: Robot Soccer World Cup IV*. Lecture Notes in Computer Science; Vol 2019, Springer, 2001, pp. 523-526.

[5] T. Bräunl, "Scaling Down Mobile Robots - A Joint Project in Intelligent Mini-Robot Research", In *Autonomous Minirobots for Research and Edutainment, Proc. 5th International Heinz Nixdorf Symposium*, U. Rückert, J. Sitte, U. Witkowski (Eds.), HNI-Verlagsshriftenreihe Germany: 2001, pp. 3-18.

[6] P. I. Corke, *Visual Control of Robots – High Performance Visual Servoing*, Somerset, England: Research Studies Press Ltd, 1996.

[7] S. Freud, "The Ego and the Id". London, Hogarth Press, 1947.

[8] M. H. Herbert, C. Thorpe, A. Stentz, Ed. *Intelligent Unmanned Ground Vehicles – Autonomous Navigation Research at Carnegie Mellon*, USA: Kluwer Academic Press, 1997.

[9] M Herman et al, “Minimalist Vision for Navigation.” In *Visual Navigation – from Biological Systems to Unmanned Ground Vehicles*, Y. Aloimonos, Ed. Mahwah, New Jersey: Lawrence Erlbaum Associates, 1997, ch. 10, pp. 275-316.

[10] T. Hamada, “Vision, Action, and Navigation in Animals.” In *Visual Navigation – from Biological Systems to Unmanned Ground Vehicles*, Y. Aloimonos, Ed. Mahwah, New Jersey: Lawrence Erlbaum Associates, 1997, ch. 2, pp. 6-25.

[11] K. Hashimoto, H. Kimura, "Visual servoing with nonlinear observer", *Proceedings IEEE International Conference on Robotics and Automation*, vol. 1 , 21-27 May 1995, pp. 484 – 489.

[12] D. Kortenkamp, “Integrating High-Speed Obstacle Avoidance, Global Path Planning, and Vision Sensing on a Mobile Robot.” In *Artificial Intelligence and Mobile Robots*, D. Kortenkamp, R. P. Bonasso, R. Murphy, Ed. USA: MIT Press, 1998.

[13] M. Kobayashi, Y. Miyamoto, K. Mitsuhashi, Y. Tanaka, "A method of visual servoing for autonomous vehicles", *Proc. 4th International Workshop on Advanced Motion Control, 1996*, vol. 1, 18-21 Mar 1996, pp. 371 –376.

[14] B. Maxwell, et al, “A Real Time Vision Module for Interactive Perceptual Agents”, *Proc. Second International Workshop, ICVS 2001, Computer Vision Systems,* Lecture Notes in Computer Science, vol. 2095, pp. 190-200, 2001.

[15] J. Petitt, T. Bräunl, "A Framework for Cognitive Agents", *International Journal of Control, Automation, and Systems* Vol. 1, No. 1, March 2003.

[16] J. A. Piepmeier, G. V. McMurray, H. Lipkin, “Tracking a Moving Target with Model Independent Visual Servoing: a Predictive Estimation Approach”, *Proc. 1998 IEEE International Conference on Robotics and Automation*, 1998, vol. 3, pp. 2652 – 2657.

[17] M. Spengler, B. Schiele, “Towards Robust Multi-cue Integration for Visual Tracking”, *Proc. Second International Workshop, ICVS 2001, Computer Vision Systems,* Lecture Notes in Computer Science, vol. 2095, pp. 93-106, 2001.

[18] W. Tianmiao, Z. Bo, “Time-varying potential field based `perception-action' behaviours of mobile robot” *Proc. IEEE ICRA*, pp.2549-2554, 1992.